

RANCANG BANGUN APLIKASI *CUTTING STOCK* *OPTIMIZATION* DENGAN METODE *INTEGER LINEAR* *PROGRAMMING*

Teguh Wijaya¹⁾ M.J. Dewiyani Sunarto²⁾ Yoppy Mirza Maulana²⁾

Program Studi/Jurusan Sistem Informasi

STMIK STIKOM Surabaya

Jl. Raya Kedung Baruk 98 Surabaya, 60298

Email : 1)teguh_wijaya31@yahoo.com, 2)dewiyani@stikom.edu, 3)yoppy@stikom.edu

Abstract: “Rejeki Glass Shop is one of the stores located in Jl . P.B. Sudirman 186 , Banyuwangi . Rejeki Glass Shop is a store that sells one of the important ingredients in the construction of a building is glass . Glass itself is very diverse , ranging from kind to the thickness of the glass .

The problem arises when the order of glass pieces found in large numbers . Order in bulk piece of glass is glass for the windows of the building or the tank . In the cutting process , often resulting inaccuracy in analyzing the pattern. As a result the rest of the glass produced less minimal . It can only be known when calculating and describing the specific , from the patterns to be cut . After the comparison is known that there ineffective and inefficiencies cutting . Problems like this are often referred to as the cutting stock problem .

Design Applications Cutting Stock Optimization Methods Integer Linear Programming In Rejeki Glass Shop, is expected to improve the cutting process becomes more effective and efficient . Effective question is about maximizing the broad field of glass , so it can be used optimally. Efficient question is the reduction in material costs that arise from the rest of the glass is wasted .”

Keywords: Cutting Stock Optimization, Integer Linear Programming.

Toko Kaca Rejeki merupakan salah satu toko yang terletak di Jl. P.B. Sudirman 186, Banyuwangi. Toko Kaca Rejeki ini merupakan toko yang menjual salah satu bahan yang penting dalam pembangunan suatu bangunan yaitu kaca. Kaca sendiri sangat beragam, mulai dari jenisnya sampai dengan ketebalan kaca tersebut.

Permasalahan muncul ketika didapati order kaca potongan dalam jumlah yang besar. Order kaca potongan dalam jumlah besar adalah kaca untuk jendela gedung atau aquarium. Pada proses pemotongan, sering terjadi ketidakteknelitian dalam menganalisa polanya. Akibatnya sisa kaca yang dihasilkan kurang minimal. Hal tersebut baru dapat diketahui ketika dilakukan perhitungan dan penggambaran secara spesifik, dari pola yang akan dipotong. Setelah dilakukan

perbandingan diketahui bahwa terjadi ketidakefektif dan ketidakefisienan pemotongan. Permasalahan seperti ini sering kali disebut sebagai *cutting stock problem*.

Dengan dibuatkannya Rancang Bangun Aplikasi *Cutting Stock Optimization* Dengan Metode *Integer Linear Programming* Pada Toko Kaca Rejeki, diharapkan dapat meningkatkan proses pemotongan menjadi lebih efektif dan efisien. Efektif yang dimaksud adalah mengenai memaksimalkan luas bidang kaca, sehingga bisa terpakai secara optimal. Efisien yang dimaksud adalah pengurangan biaya material yang muncul dari sisa kaca yang terbuang.

METODE

Cutting Stock Optimization

Cutting stock optimization merupakan permasalahan optimasi dalam pengkombinasian, sehingga dapat ditentukan solusi dari beberapa solusi yang mungkin, yang memenuhi fungsi pembatas yang ada. Solusi yang ditawarkan adalah dengan mengkombinasikan beberapa pieces dengan ukuran berbeda ke dalam persegi empat (bahan baku) sehingga didapatkan sisa kaca seminimal mungkin.

Karakteristik Pemotongan Bahan (*Cutting Stock*)

Karakteristik pemotongan bahan adalah sebagai berikut:

1. Terdapat bahan baku yang berbentuk persegi empat yang mempunyai ukuran tertentu.
2. Terdapat m jenis potongan yang dihasilkan (yang selanjutnya disebut dengan *pieces*) yang masing-masing berukuran n dengan jumlah permintaan o tertentu.
3. Setiap potong mempunyai nilai tertentu yang bisa berupa keuntungan yang diperoleh atau berupa ukuran luas dalam upaya meminimasi sisa bahan baku.
4. Berusaha membentuk suatu *layout* potong yang meminimumkan fungsi tujuan yang melekat pada setiap potong yang ada.

Integer Linear Programming

Linear Programming merupakan metode atau teknik matematik yang digunakan untuk membantu dalam pengambilan keputusan. Di dalam *linear programming*, seluruh fungsinya (fungsi objektif serta fungsi pembatas) haruslah linear.

Terdapat empat asumsi dasar dalam penyelesaian masalah dengan menggunakan model *linear programming*, yaitu : Menurut Lieberman dkk (1995 : 38-44)

1. *Proporsionality*.
2. *Divisibility*.
3. *Addivity*.
4. *Certainty* .

Integer Programming (IP) merupakan bentuk lain dari *Linear Programming* (LP) yang muncul karena tidak semua variabel keputusan dapat berupa bilangan pecahan dengan kata lain asumsi *divisibility* melemah atau hilang sama sekali.

Metode yang digunakan untuk memaksa pemecahan optimum dari *linear programming* yang dilonggarkan untuk bergerak ke arah pemecahan *integer* yang diinginkan adalah *branch & bound*. Algoritma *Branch & Bound* berlaku baik untuk masalah *integer* murni maupun masalah *integer* campuran. Keuntungan utamanya adalah bahwa batas atas tersebut dapat diestimasi dengan cepat dan dengan perhitungan minimal.

Pengembangan Model Sistematis

Pengembangan model matematis dapat dimulai dengan menjawab ketiga pertanyaan berikut : Menurut Hamdy (1996 : 17)

1. Apa yang diusahakan untuk ditentukan oleh model tersebut? Dengan kata lain, apa variable (yang tidak diketahui) dari masalah tersebut?
2. Apa batasan yang harus dikenakan atas variable untuk memenuhi batasan sistem yang dimodel tersebut?
3. Apa tujuan (sasaran) yang harus dicapai untuk menentukan pemecahan optimum (terbaik) dari semua nilai yang layak dari variable tersebut?

Software Development Life Cycle

Siklus Hidup Pengembangan Sistem atau *Software Development Life Cycle* (SDLC) adalah proses mengembangkan atau mengubah suatu sistem perangkat lunak dengan menggunakan model-model dan metodologi yang digunakan orang untuk mengembangkan sistem-sistem perangkat lunak sebelumnya.

Ada beberapa tahapan dalam SDLC yaitu antara lain:

1. *Software Requirement*
 - Elisitasi Kebutuhan
 - Analisis Kebutuhan
 - Spesifikasi Kebutuhan
2. *Software Design*
3. *Software Construction*
4. *Software Testing*
5. *Software Maintenance*

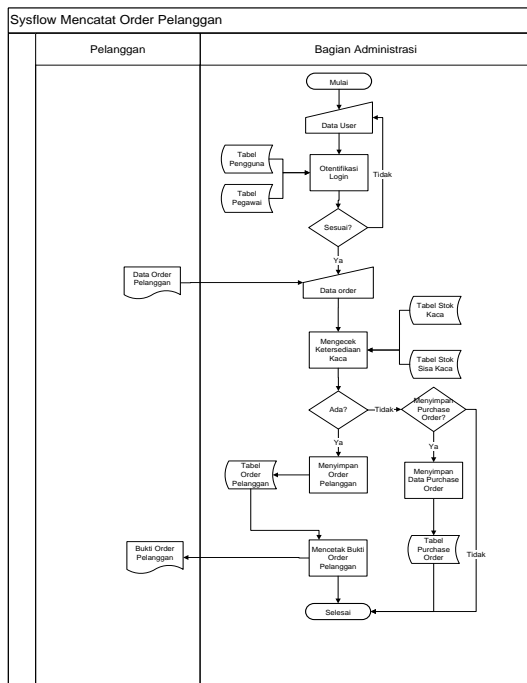
HASIL DAN PEMBAHASAN

Alur Sistem Baru

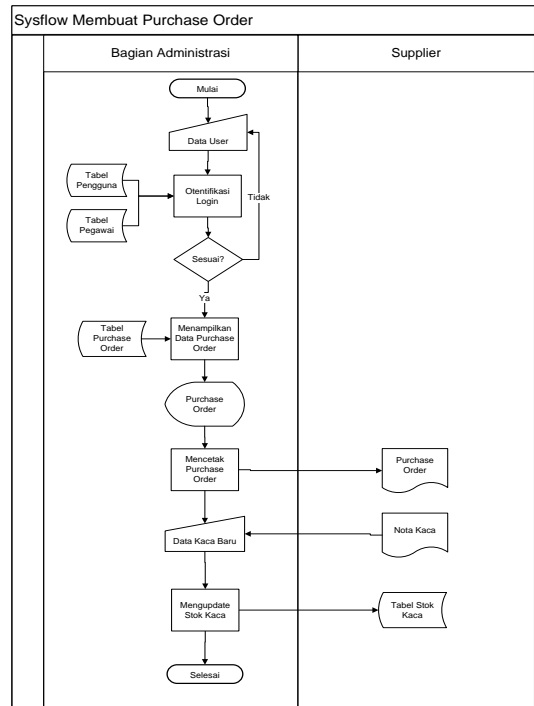
Dalam membangun suatu perangkat lunak, dilakukan proses desain berdasarkan kebutuhan masing – masing pengguna. Dikatakan sesuai dengan kebutuhan pengguna karena proses desain tersebut dilakukan berdasarkan hasil analisis kebutuhan yang telah dilakukan

sebelumnya. Adapun rancangan perangkat lunak tersebut berupa rancangan alur sistem (*System Flow*), *Context Diagram*, *Data Flow Diagram*, dan rancangan tampilan antar muka (*Design Interface*).

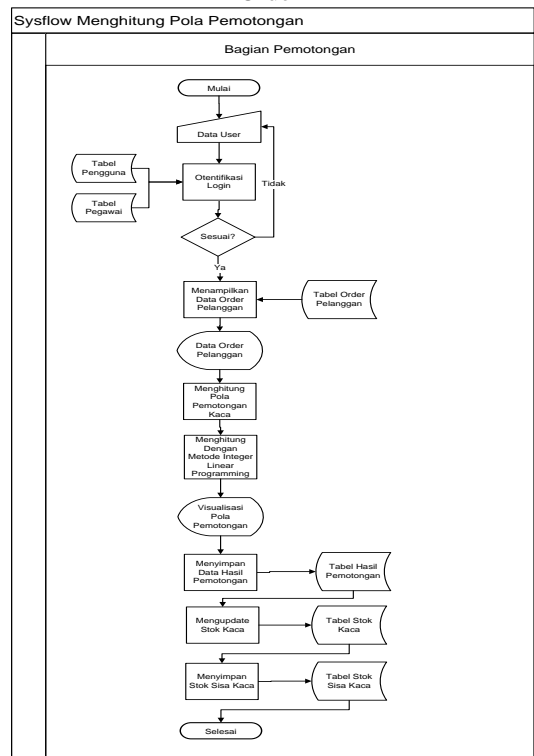
Sesuai dengan hasil analisis kebutuhan, telah didefinisikan bahwa pengguna yang menggunakan sistem secara langsung berjumlah 2 (dua) pengguna yaitu Bagian Administrasi dan Bagian Pemotongan, sehingga perlu digambarkan rancangan alur sistem yang baru untuk tiap pengguna tersebut.



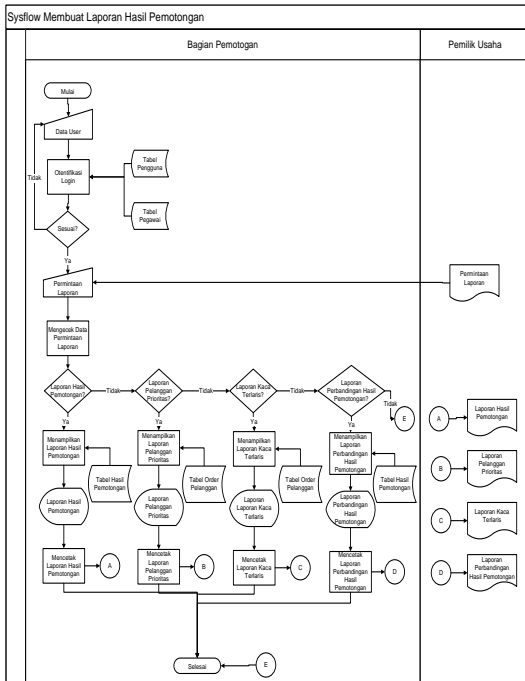
Gambar 1. Alur Sistem Mencatat Order Pelanggan



Gambar 2. Alur Sistem Membuat Purchase Order



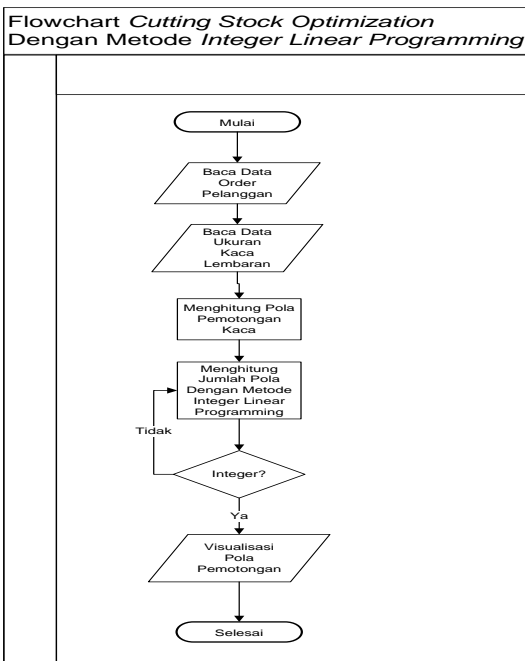
Gambar 3. Alur Sistem Menghitung Pola Pemotongan.



Gambar 4. Alur Sistem Membuat Laporan Hasil Pemotongan

Flowchart Program

Flowchart dari program Rancang Bangun Aplikasi *Cutting Stock Optimization* Dengan Metode *Integer Linear Programming* dapat dilihat pada Gambar 5.

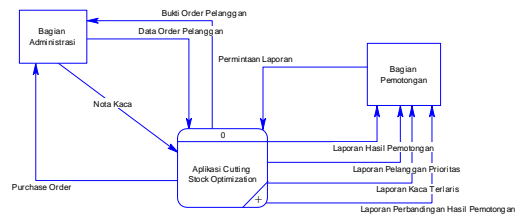


Gambar 5. Flowchart Program.

Data Flow Diagram

Context Diagram

Context Diagram dari aplikasi *cutting stock optimization* ini terdiri dari 2 (dua) entitas yaitu Bagian Administrasi dan Bagian Pemotongan.



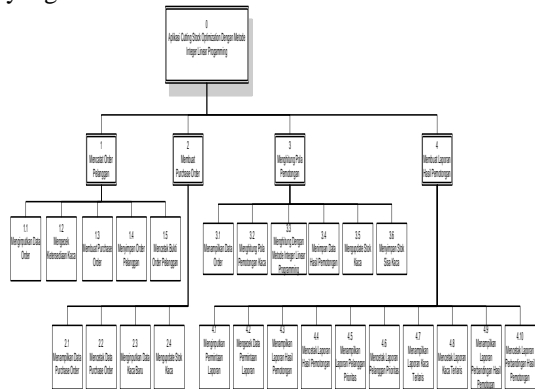
Gambar 6. Context Diagram

Bagian Administrasi bertugas sebagai orang yang mencatat order – order pelanggan. Selain itu bagian administrasi juga bertugas untuk membuat purchase order, ketika stok kaca di gudang telah habis.

Bagian Pemotongan berperan sebagai orang yang menghitung kebutuhan pemotongan kaca dan juga mengeksekusi pemotongan kaca.

Diagram Berjenjang

Diagram Berjenjang merupakan diagram yang menggambarkan pembagian fungsi-fungsi dari sistem menjadi sub sistem yang lebih kecil.

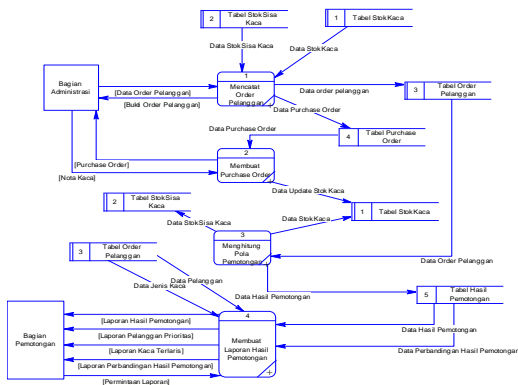


Gambar 7. Diagram Berjenjang

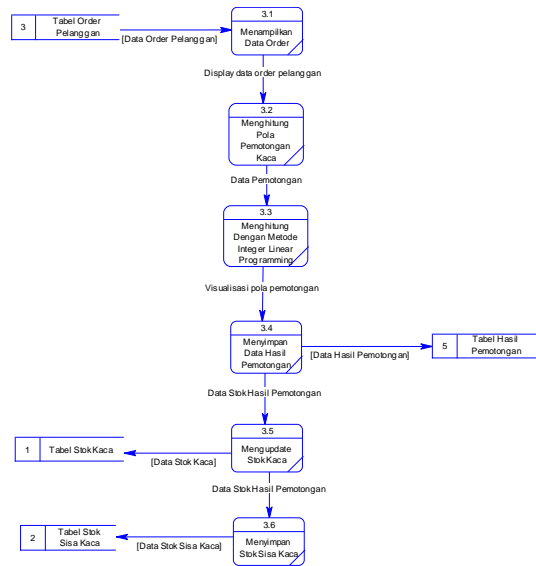
DFD Level 0

Setelah membuat *context diagram* dan diagram berjenjang, perancangan dilanjutkan dengan membagi *context diagram* menjadi proses-proses yang lebih kecil dan rinci sesuai dengan diagram berjenjang. Adapun *data flow*

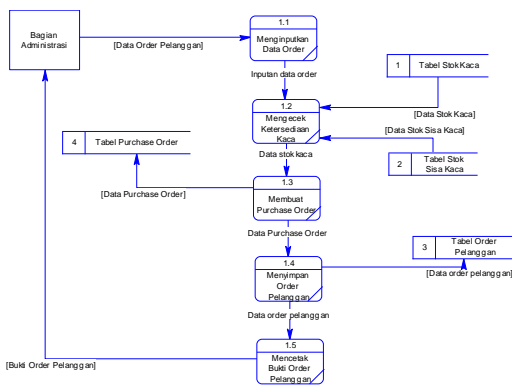
diagram level 0 aplikasi *cutting stock optimization* dengan metode *integer linear programming* seperti terlihat pada gambar-gambar berikut ini.



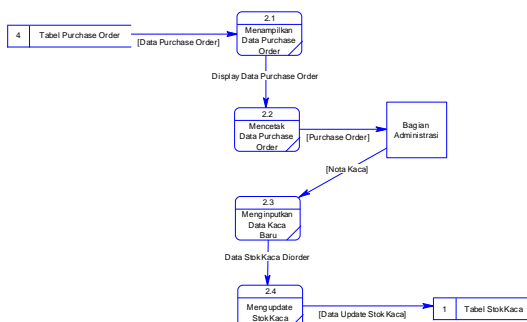
Gambar 8. DFD Level 0



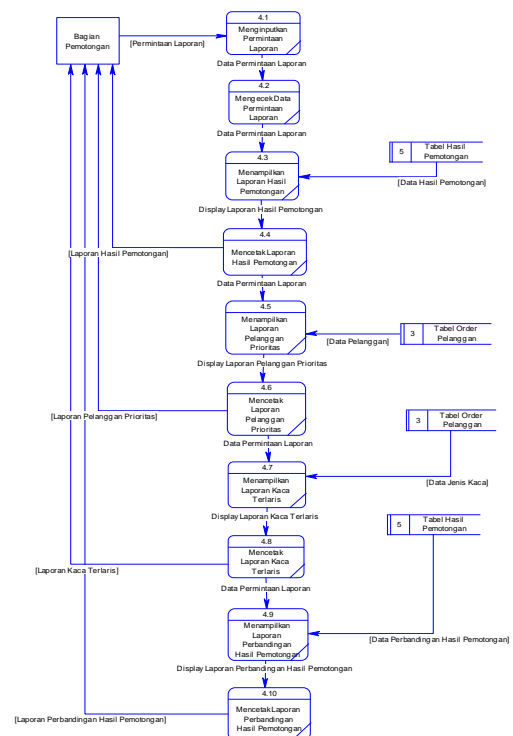
Gambar 11. DFD Level 1 Menghitung Pola Pemotongan



Gambar 9. DFD Level 1 Mencatat Order Pelanggan



Gambar 10. DFD Level 1 Membuat Purchase Order



Gambar 12. DFD Level 1 Membuat Laporan Hasil Pemotongan

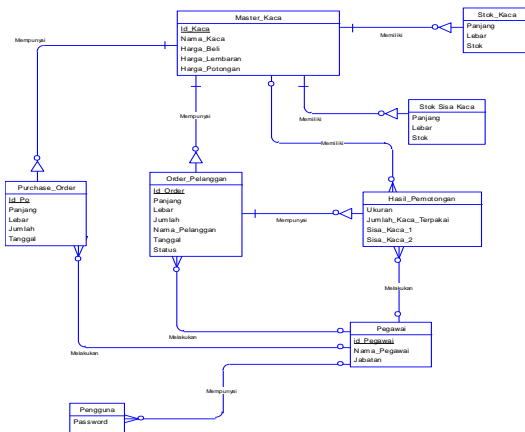
Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) digunakan untuk mengidentifikasi data yang akan diambil, disimpan dan diambil kembali (*retrieve*) untuk keperluan-keperluan tertentu dalam mendukung kegiatan yang dilakukan oleh

sistem. ERD juga digunakan untuk mengidentifikasi asal data yang dibutuhkan dan dilaporkan.

Dalam aplikasi *cutting stock optimization* dengan metode *integer linear programming* ini, disajikan dalam bentuk *Conceptual Data Model (CDM)* dan *Physical Data Model (PDM)*. Gambar 13 merupakan *Conceptual Data Model* dari aplikasi yang akan dikembangkan.

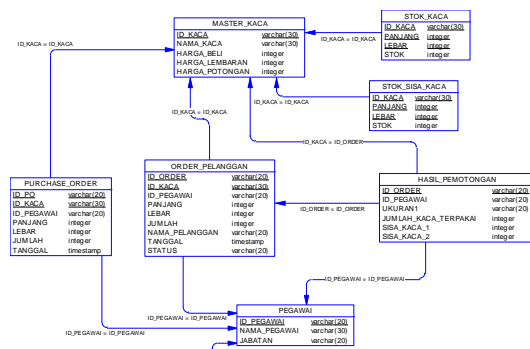
Conceptual Data Model (CDM)



Gambar 13. Conceptual Data Model (CDM)

Physical Data Model (PDM)

Dengan melakukan *generate Conceptual Data Model (CDM)* diatas, maka akan diperoleh *Physical Data Model(PDM)*. Gambar 9 merupakan *Physical Data Model* dari aplikasi yang akan dikembangkan.



Gambar 14. Physical Data Model (PDM)

IMPLEMENTASI

Form Login

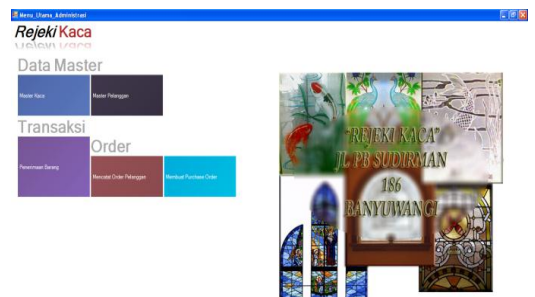
Form Login merupakan form ditampilkan pertama kali saat aplikasi dijalankan dan berlaku untuk semua pengguna. *Form Login* dibutuhkan untuk membedakan hak akses pengguna sesuai dengan otoritasnya masing-masing.. *Form Login* untuk setiap pengguna dapat dilihat seperti pada Gambar 15.



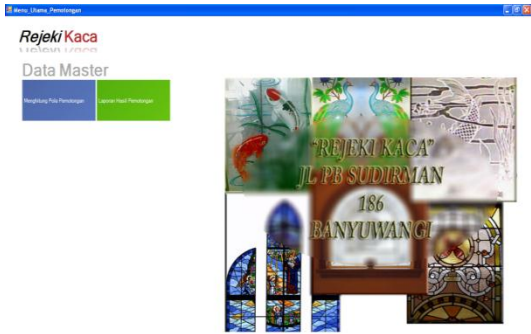
Gambar 15. Tampilan menu Login

Form Menu Utama

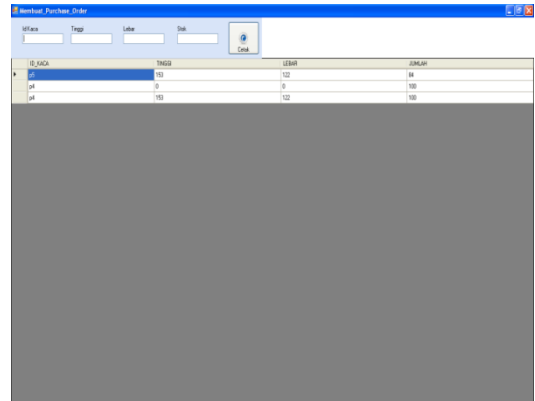
Form Menu Utama adalah form yang muncul ketika user telah berhasil melakukan proses login. Di dalam *Form Menu Utama* terdapat berbagai pilihan menu untuk memunculkan form-form selanjutnya yang dibutuhkan. Tampilan dari *Menu Form Utama* terdiri dari 2, yaitu *Form Menu Utama Administrasi* dan *Form Menu Utama Pemotongan* dapat dilihat pada Gambar 16 dan Gambar 17:



Gambar 16. Tampilan Menu Utama Administrasi



Gambar 17. Tampilan Menu Utama Pemotongan

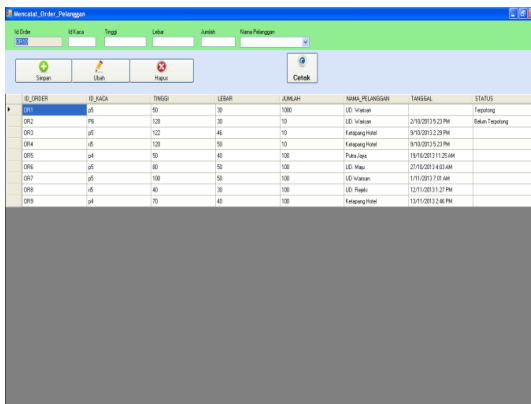


Gambar 19. Tampilan menu Membuat Purchase Order

Form Aplikasi Pengguna Sebagai Administrasi

a) Form Menu Mencatat Order Pelanggan

Menu Mencatat Order Pelanggan digunakan oleh bagian Administrasi untuk menambahkan data order pelanggan dan juga mengubah / menghapus data order pelanggan lama. Data order pelanggan digunakan sebagai inputan awal untuk selanjutnya diolah kedalam perhitungan pola pemotongan. Tampilan Form Mencatat Order Pelanggan dapat dilihat seperti tampak pada Gambar 18.



Gambar 18. Tampilan menu Mencatat Order Pelanggan

b) Form Menu Membuat Purchase Order

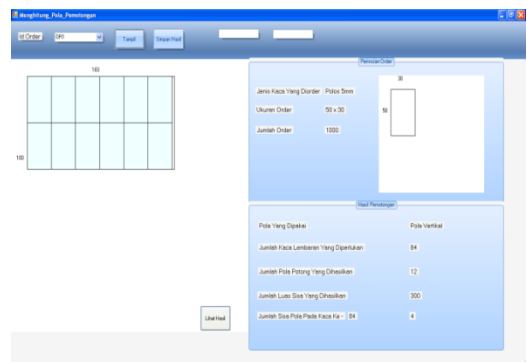
Menu Membuat Purchase Order digunakan oleh bagian Administrasi untuk mencetak data purchase order yang akan diberikan kepada supplier. Data *purchase order* ini berisi jumlah dan jenis kaca yang perlu diorder kepada supplier untuk selanjutnya dilakukan proses pemotongan. Tampilan Form Menu Membuat Purchase Order dapat dilihat seperti tampak pada Gambar 19.

Pengguna Aplikasi Pengguna Sebagai Bagian Pemotongan

Bagian Pemotongan merupakan user yang melakukan proses pemotongan setelah order pelanggan dicatat. Bagian Pemotongan bertugas untuk mengolah data order pelanggan, untuk selanjutnya dilakukan proses pemotongan kaca secara real. Adapun tampilan form untuk Bagian Pemotongan adalah sebagai berikut:

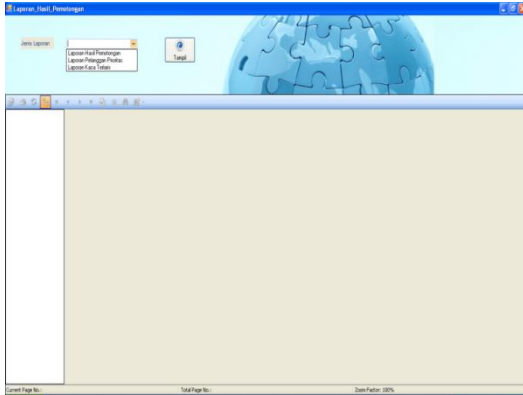
a) Form Menghitung Pola Pemotongan

Form Menghitung Pola Pemotongan merupakan form yang digunakan untuk mengolah data order pelanggan yang telah dicatat menjadi data pemotongan kaca yang lebih terperinci. Tujuannya adalah untuk mempermudah bagian pemotongan dalam memvisualisasikan pola pemotongan, ukuran kaca yang paling ideal, jumlah kaca yang diperlukan untuk selanjutnya diaplikasikan dalam proses pemotongan yang real. Adapun tampilan form Menghitung Pola Pemotongan seperti tampak pada Gambar 20.



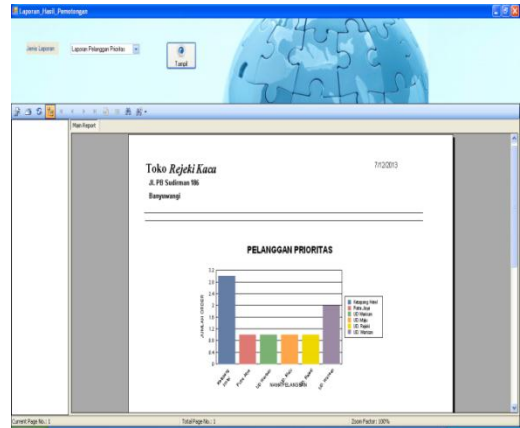
Gambar 20. Tampilan menu Menghitung Pola Pemotongan

b) Form Membuat Laporan Hasil Pemotongan
 Form Membuat Laporan Hasil Pemotongan berguna bagi Bagian Pemotongan untuk melihat dan memperinci hasil proses pemotongan yang dilakukan untuk selanjutnya dapat dicetak dan disajikan kepada pimpinan toko. Hasil laporan proses pemotongan berguna untuk memberikan informasi-informasi verbal yang menunjukkan perincian proses pemotongan. Adapun tampilan form Menu Report seperti tampak pada Gambar 21.

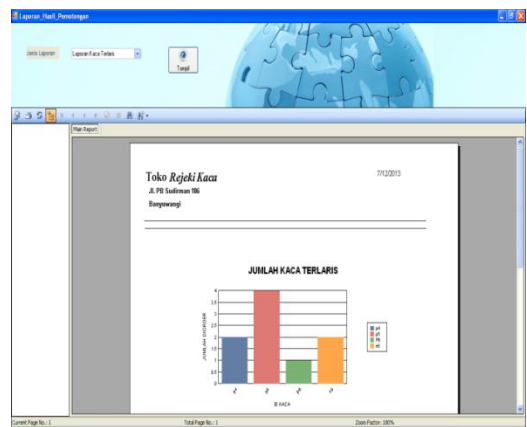


Gambar 21. Tampilan menu Report

Adapun hasil proses *report* yang dipilih berdasarkan Gambar 21 dapat dilihat pada gambar-gambar berikut ini:



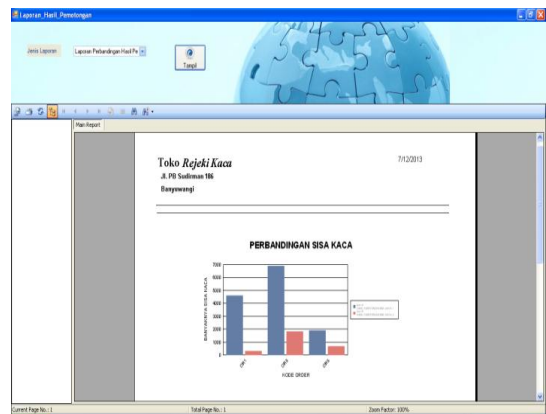
Gambar 23. Laporan Pelanggan Prioritas



Gambar 24. Laporan Kaca Terlaris

NO. ORDER	NO. PEMOTONGAN	JMLAH	JMLAH KACA TERBUK
001	00001	1000	1000

Gambar 22. Laporan Hasil Pemotongan



Gambar 25. Laporan Perbandingan Sisa Kaca

Kesimpulan

Setiap penelitian tentu menghasilkan suatu kesimpulan yang dapat digunakan untuk memperkuat hasil penelitian. Adapun kesimpulan yang dapat diambil dari penelitian ini adalah sebagai berikut:

1. Luas bidang kaca yang dipakai dapat dioptimalkan dengan memilih pola yang telah dihitung menggunakan metode *integer linear programming*.
2. Biaya material yang muncul dari luas sisa kaca yang terbuang, dapat dikurangi menjadi sekitar 56,45%.

Saran

Adapun saran yang dapat diberikan pada penelitian ini adalah sebagai berikut:

1. Aplikasi ini memerlukan aplikasi yang dapat menampilkan gambar visualisasi pemotongan dengan lebih fleksibel dan inovatif.
2. Penelitian ini dapat dikembangkan lebih lanjut dengan menambahkan metode lainnya yang terkait seperti algoritma genetika

RUJUKAN

Dimiyati, Tjutju Tarlih dan Ahmad Dimiyati, *Operations Research : Model-model Pengambilan Keputusan*, Sinar Baru Algensindo : Bandung, 2003

England, John Wiley & Sons. IEEE. " Guide to the Software Engineering Body of Knowledge 2004 Version:” *SWEBOK A Project of the IEEE Computer Society Professional Practices Committee*. Tahun 2004.

Hamdy A Taha, *Riset Operasi jilid 1*, Binarupa Aksara, Jakarta, 1996

Kristanto, H, *Konsep dan Perancangan Database*, Andi Offset, Yogyakarta, 1994.

Lieberman, Gerald J. and Frederick S, Hillier, *Introduction to Operations Research-6th ed*, McGraw Hill : New York, 1995

Royce, Winston (1970), "Managing the Development of Large Software Systems", Proceedings of IEEE WESCON 26 (August): 1–9.

Scheithauer, G and G. Belov, *A Branch-and-Cut-and-Price Algorithm for One-Dimensional Stock Cutting and Two-Dimensional Two-Stage Cutting*, Technical Report MATH-NM-03, Dresden University, 2003

Simatupang, Togar M., *Pemodelan Sistem*, Nindita : Klaten, 1995

Sommerville, I. and Sawyer, P. (1997). *Requirements Engineering: A Good Practice*, Chichester

Yuswanto, "Pemrograman Dasar Visual Basic .NET 2005", Jakarta, 2008.