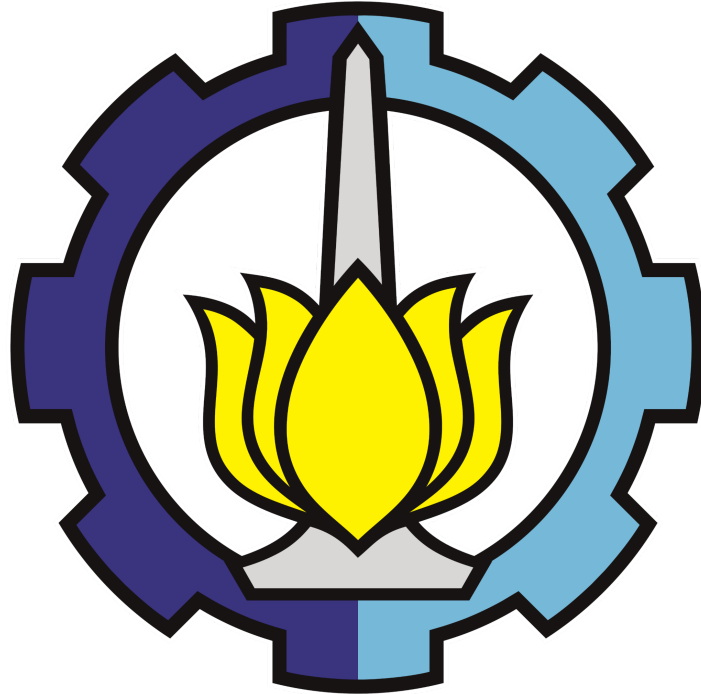


Laporan Tugas Besar
Basis Data Kelas B Semester Gasal 2023/2024



Dosen Pengampu :
T. Dwi Ary Widhianingsih, Ph.D
Adatul Mukarromah, S.Si, M.Si

Disusun oleh :
Kelompok 3 Basis Data-B
Bayuaji Wisantoko (5003221024)
Muhammad Aflah Khozi Susanto (5003221074)
Muhammad Ilham Ramadhan (5003221185)

DEPARTEMEN STATISTIKA
FAKULTAS SAINS DAN ANALITIKA DATA
INSTITUT TEKNOLOGI SEPULUH NOPEMBER SURABAYA
2023

BAB I

PENDAHULUAN

1.1 Latar Belakang

Pengelolaan informasi yang efektif dan efisien merupakan aspek krusial dalam menjalankan suatu bisnis, terutama dalam operasional toko. Pemilik toko menghadapi tantangan serius terkait manajemen data dan informasi pada sistem sebelumnya. Keluhan pelanggan yang muncul adalah lambatnya tim sales atau karyawan toko dalam memberikan informasi dasar mengenai produk yang dijual, seperti jumlah stok, pilihan warna, informasi spesifikasi, dan aspek-aspek penting lainnya.

Akar permasalahan utama terletak pada sistem pencatatan data yang belum terorganisir dengan baik dan belum dimanfaatkan secara optimal. Kesadaran pemilik toko atas ketidakefektifan ini menciptakan dampak langsung pada kinerja tim sales, yang dapat mengurangi kepuasan pelanggan. Oleh karena itu, pemilik toko memutuskan untuk melakukan perombakan pada sistem basis data yang menjadi landasan informasi.

Langkah strategis yang diambil adalah membangun sistem basis data yang lebih efisien dan terstruktur. Pembaruan ini diharapkan dapat meningkatkan kecepatan dan ketepatan mengenai ketersediaan informasi produk, serta menciptakan pengalaman berbelanja yang lebih memuaskan bagi pelanggan. Keputusan ini sebagai upaya meningkatkan optimalitas sistem kerja secara keseluruhan, menciptakan lingkungan yang lebih efisien, dan memastikan pertumbuhan bisnis yang berkelanjutan.

Dalam implementasinya, pemilihan MySQL sebagai sistem manajemen basis data menjadi langkah penting. Penggunaan MySQL diharapkan dapat memberikan struktur yang lebih baik pada pencatatan data, meningkatkan efisiensi akses informasi, dan mendukung transformasi positif dalam operasional toko. Dengan ini, pemilik toko tidak hanya mengantisipasi permasalahan yang ada, tetapi juga mengambil langkah proaktif untuk mengoptimalkan pengelolaan informasi dan memastikan kemajuan berkelanjutan dalam bisnisnya.

1.2 Rumusan Masalah

Berdasarkan latar belakang di atas, dapat dirumuskan sebuah rumusan masalah sebagai berikut:

- a. Bagaimana merancang sistem basis data toko yang dapat memudahkan pelanggan dalam melakukan proses pembelian barang?
- b. Bagaimana merancang sistem basis data toko yang dapat memudahkan karyawan toko dalam melakukan pendataan stok barang maupun pendataan pembelian oleh pelanggan?

1.3 Tujuan

Berdasarkan rumusan masalah di atas, manfaat dari analisis ini adalah sebagai berikut:

- a. Menganalisis kebutuhan pelanggan dan merancang sistem basis data toko yang dapat meningkatkan efisiensi dan kemudahan dalam proses pembelian barang.
- b. Menciptakan sistem basis data toko yang memfasilitasi karyawan dalam mengelola stok barang dan mencatat transaksi pembelian oleh pelanggan secara efisien.

BAB II

ANALISIS DAN PEMBAHASAN

2.1 Ruang Lingkup Basis Data

Toko elektronik menjadi salah satu model usaha yang semakin meluas di era digital saat ini. Fenomena ini dipicu oleh kemajuan pesat dalam teknologi dan kesadaran yang meningkat di kalangan masyarakat terhadap pentingnya perangkat elektronik. Dalam lingkup toko elektronik ini, pelanggan dapat menemukan berbagai produk, termasuk televisi, lemari es, vacuum cleaner, AC, rice cooker, dan blender, yang semuanya merupakan perangkat elektronik yang memudahkan kehidupan sehari-hari.

Toko elektronik ini menawarkan beragam pilihan produk untuk memenuhi kebutuhan konsumen. Misalnya, dalam kategori AC, konsumen dapat memilih dari merek ternama seperti Samsung, Daikin, Panasonic, dan Sharp, yang masing-masing menyediakan berbagai tipe dengan beragam spesifikasi dan warna. Sementara itu, untuk televisi, toko ini menyediakan berbagai opsi dari merek terkenal seperti Samsung, Xiaomi, LG, Sharp, dan Sony. Setiap merek menawarkan tipe-tipe TV yang berbeda, memberikan konsumen kebebasan untuk memilih sesuai dengan preferensi mereka, baik dari segi spesifikasi maupun warna.

Dalam kategori rice cooker, tersedia berbagai pilihan dari merek ternama seperti Dyson, Electrolux, Bosch, Panasonic, dan Samsung. Konsumen dapat memilih rice cooker dengan tipe dan fitur yang sesuai dengan kebutuhan memasak mereka. Untuk blender, toko ini menyajikan opsi dari merek Oxone, Miyako, Philips, Cosmos, dan Sharp. Setiap merek menawarkan blender dengan berbagai fitur dan desain yang dapat disesuaikan dengan kebutuhan dapur konsumen.

Lemari es hadir dalam berbagai pilihan dari merek LG, Samsung, Sharp, Toshiba, dan Polytron. Konsumen dapat memilih lemari es dengan tipe yang sesuai dengan kapasitas dan kebutuhan penyimpanan mereka, sambil memilih warna yang cocok dengan desain dapur. Tidak ketinggalan, vacuum cleaner juga ditawarkan dalam berbagai varian dari merek Dyson, Electrolux, Bosch, Panasonic, dan Samsung. Setiap merek menyajikan vacuum cleaner dengan spesifikasi yang beragam untuk memastikan konsumen dapat menemukan yang sesuai dengan kebutuhan pembersihan rumah mereka.

Semua produk ini memberikan konsumen banyak opsi, baik dari segi merek, tipe, warna, maupun spesifikasi, sehingga mereka dapat memilih produk yang paling sesuai dengan kebutuhan dan preferensi mereka.

2.2 Normalisasi Basis Data

Normalisasi tabel adalah proses penataan ulang tabel dalam basis data untuk menghilangkan redundansi data dan memastikan ketergantungan data yang benar. Normalisasi terdiri dari tiga tingkatan, yaitu 1NF, 2NF, dan 3NF.

2.2.1 Tabel 1NF

Bentuk normal pertama (1NF) adalah bentuk normal yang memenuhi dua syarat, yaitu tidak ada data duplikat dalam satu baris, dan data disimpan dalam bentuk tabel dua dimensi. Pada tahap normalisasi pertama, transaksi yang membeli dua atau

lebih produk yang berbeda akan disimpan dalam baris yang berbeda pada tabel Detail Transaksi.

2.2.2 Tabel 2NF

Bentuk normal kedua (2NF) adalah bentuk normal yang memenuhi dua syarat, yaitu memenuhi bentuk normal pertama (1NF) dan semua atribut non-primary key memiliki hubungan fungsional dengan primary key. Untuk identifikasi primary key, pada tabel Detail Transaksi (1NF) terdapat 6 primary key berupa ID Payment, ID Kasir, ID Produk, ID Pelanggan, ID Pengiriman, dan ID Membership. Sementara ID Transaksi tidak bersifat unik karena terdapat ID yang sama namun memiliki record yang berbeda. Pada tahap normalisasi kedua, tabel Detail Transaksi dipecah menjadi 7 tabel baru, yaitu Tabel Payment, Kasir, Produk, Pelanggan, Pengiriman, Membership, dan Transaksi. Kemudian, Hubungan antara tabel asal dan tabel baru adalah dengan menggunakan foreign key sehingga menjadi Tabel Detail Transaksi baru (2NF).

2.2.3 Tabel 3NF

Bentuk normal ketiga (3NF) adalah bentuk normal yang memenuhi dua syarat, yaitu memenuhi bentuk normal kedua (2NF) dan tidak ada atribut non-primary key yang bergantung pada atribut non-primary key lainnya. Pada tabel 3NF ini, dari tabel Detail Transaksi (2NF) kita memisahkan atribut yang bergantung parsial dan menyesuaikan nama atributnya. Produk yang dipesan bergantung dengan ID Produk sehingga produk yang dipesan cukup di munculkan dengan ID Produk saja.

2.3 Tabel Basis Data

Dalam membangun sistem basis data *MySQL*, struktur data menjadi pondasi penting. Penyusunan struktur tabel bertujuan untuk menyusun data dengan teratur, sesuai tempatnya, dan menghubungkan antar tabelnya. Berikut adalah struktur dari tiap-tiap tabel yang telah disusun di *MySQL*.

Pada tabel kasir, menyimpan informasi mengenai nama, tanggal lahir, dan juga ID dari karyawan kasir. Pada tabel ini, ID Kasir menjadi *primary key* yang mengidentifikasi secara unik setiap karyawan kasir. Kemudian pada tabel produk, terdapat informasi mengenai ID Produk yang mengidentifikasi secara unik (*primary key*) data-data pada tabel ini yang mencakup nama, kategori, merk, spesifikasi, warna, stok, dan harga dari masing-masing produk. Pada tabel *membership*, terdapat informasi mengenai Id *membership* sebagai *primary key* yang mengidentifikasi jenis *membership*, diskon, garansi, dan fee untuk tiap jenis *membership*. Pada tabel ini memuat diskon dan garansi yang didapat pelanggan apabila memiliki *membership* tertentu serta fee yang harus dibayarkan untuk mendapatkan suatu *membership*.

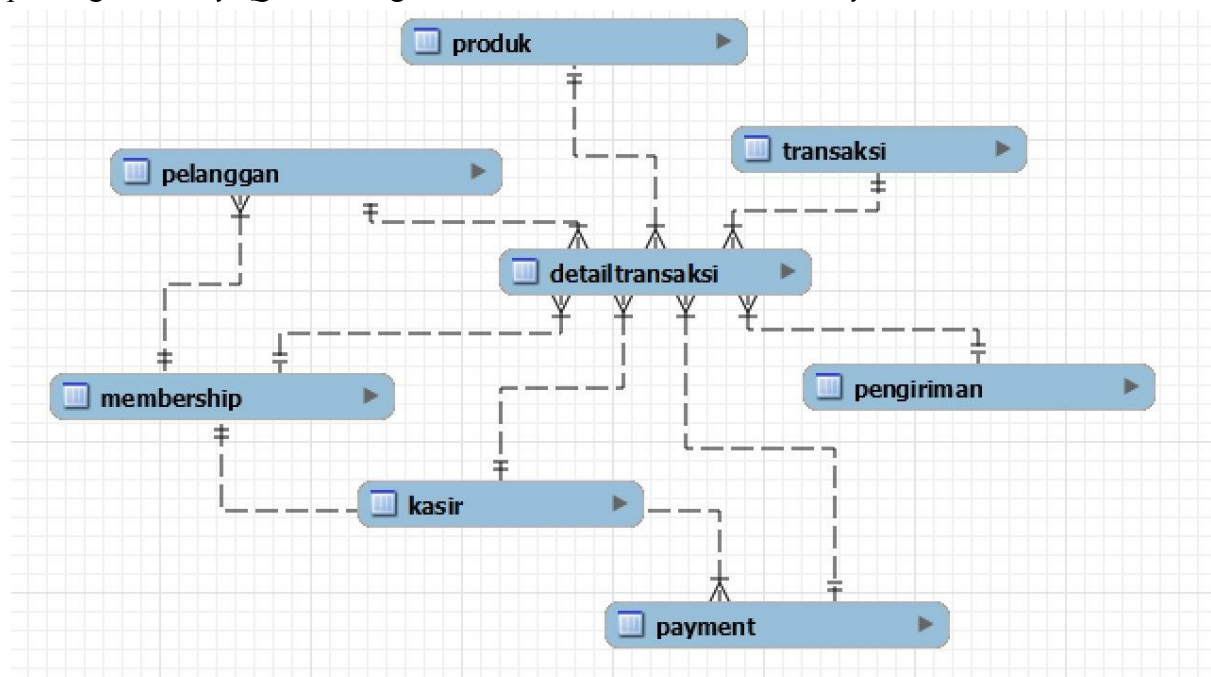
Pada tabel pelanggan memuat 2 ID yaitu, ID pelanggan sebagai primary key dan ID *membership* yang terhubung dengan ID *membership* di tabel *membership* (*foreign key*). Selain itu, tabel ini juga memuat informasi mengenai nama, alamat, nomor telepon, email, dan juga jarak alamat pelanggan ke toko. Kemudian, terdapat tabel transaksi yang memuat ID transaksi sebagai *primary key* untuk mengidentifikasi tanggal dan waktu transaksi dilakukan oleh pelanggan. Pada tabel *payment*, terdapat informasi mengenai jenis *membership*, *payment*

method, dan bunga yang didapatkan sesuai dengan jenis *membership* dari pelanggan. Dalam tabel ini terdapat ID *payment*, yang secara unik mengidentifikasi data-data di tabel ini. Jenis *membership* pada tabel ini ber *foreign key* dengan jenis *membership* pada tabel *membership*.

Tabel pengiriman, memuat data-data mengenai ID pengiriman sebagai *primary key* yang mengidentifikasi secara unik mengenai jasa pengiriman, jenis pengiriman, dan juga biaya pengiriman. Pada tabel detail transaksi, memuat informasi dari tabel lain sesuai dengan transaksi yang dilakukan dengan isi tabel berupa ID transaksi, ID kasir, ID pelanggan, ID *membership*, ID produk, ID pengiriman, ID *payment*, dan juga jumlah produk yang dibeli oleh pelanggan. Dalam tabel ini setiap ID ber *foreign key* dengan tabel lainnya, seperti ID kasir dengan tabel ID kasir pada tabel kasir. Pada tabel ini tidak memuat *primary key* karena ID transaksi bisa muncul dua kali untuk tiap transaksi dengan produk lebih dari satu.

2.4 Relasi Antar Tabel Basis Data

Dalam sistem basis data, terdapat 3 macam *relationship* atau hubungan antar tabel, yaitu *One-to-One*, *One-to-Many*, dan *Many-to-Many*. Relasi *One-to-One* adalah hubungan antar tabel dimana dalam satu kondisi yang menjelaskan bahwa satu baris tabel pertama hanya berhubungan dengan satu baris tabel kedua. Relasi *One-to-Many* hubungan antar tabel dimana dalam satu kondisi yang menjelaskan bahwa satu baris tabel pertama berhubungan dengan banyak baris tabel kedua dan begitu pula sebaliknya. Relasi *Many-to-Many* adalah hubungan antar tabel dimana dalam satu kondisi yang menjelaskan bahwa banyak baris tabel pertama dapat berhubungan dengan banyak baris tabel kedua. Berikut adalah tampilan *Entity Relationship Diagram (ERD)* untuk *database* toko yang telah kami susun tanpa dilakukan perubahan terkait garis relasi antar tabel, yang secara umum diatur oleh aplikasi pemrograman *MySQL* adalah garis relasi untuk relasi *One-to-Many*.



Berdasarkan tampilan tangkapan layar *Entity Relationship Diagram (ERD)* tersebut diketahui bahwa seluruh tabel selain tabel *Detailtransaksi* berelasi dengan dan atau melalui tabel *Detailtransaksi*. Untuk tabel *Pelanggan* dengan *Produk* berelasi dengan hubungan relasi

Many-to-Many dengan *junction table* Detailtransaksi, sehingga pelanggan dapat membeli lebih dari satu produk dan produk dapat dibeli oleh lebih dari satu pelanggan. Maka, dengan relasi tersebut dapat diketahui bahwa antara tabel Produk dan tabel Transaksi juga berelasi *Many-to-Many* dengan *junction table* Detailtransaksi, sehingga dalam transaksi dapat mencakup lebih dari satu produk dan produk dapat dicakup di dalam lebih dari satu transaksi. Begitu pula dengan hubungan antara tabel Pelanggan dan Transaksi yaitu berelasi *Many-to-Many* dengan *junction table* Detailtransaksi.

Selanjutnya, terdapat relasi antara tabel Pelanggan dengan tabel *Membership* yaitu relasi *One-to-Many* yaitu ketika satu *ID Membership* dapat digunakan oleh banyak ID Pelanggan, sedangkan ID Pelanggan hanya bisa memiliki satu *ID Membership*. Selanjutnya, tabel *Membership* juga memiliki relasi dengan tabel *Payment* yaitu berupa relasi *One-to-Many* yaitu satu *ID Membership* dapat digunakan pada lebih dari satu *ID Payment*, sedangkan satu *ID Payment* hanya dapat memiliki satu *ID Membership*. Berikutnya, tabel Kasir memiliki hubungan *One-to-Many* terhadap tabel Detailtransaksi yaitu satu ID Kasir dapat digunakan pada lebih dari satu ID Transaksi, namun satu ID Transaksi hanya dapat memiliki satu ID Kasir. Selanjutnya, tabel Pengiriman memiliki hubungan *One-to-Many* terhadap tabel Detailtransaksi yaitu satu ID Pengiriman dapat digunakan pada lebih dari satu ID Transaksi, namun satu ID Transaksi hanya dapat memiliki satu ID Pengiriman.

2.5 Alur Kerja Basis Data

Dalam basis data berikut, terdapat dua alur utama, yaitu alur pembelian konsumen atau alur penjualan toko dan alur penambahan atau pemasokan stok. Berikut penjelasan terkait masing-masing alur tersebut.

2.5.1 Alur Pembelian Pelanggan atau Alur Penjualan Toko

Dalam basis data berikut, Pelanggan yang memiliki ID Pelanggan dan ID *Membership* membeli produk sesuai yang diinginkan. Kemudian, produk yang diinginkan akan dimasukkan ke dalam transaksi dengan pembeda berupa ID Transaksi yang setiap transaksinya akan dibantu oleh Kasir dengan masing-masing kasir memiliki ID Kasir yang berbeda antara satu sama lain. Setelah itu, pelanggan akan memilih mengirimkan produk sesuai dengan jasa dan jenis pengiriman yang diinginkan. Setelah transaksi berjalan, maka stok produk yang dibeli oleh pelanggan akan berkurang dari stok produk yang dimiliki oleh toko yang mana masing-masing produk di toko memiliki stok produk awal sejumlah 25.

2.5.2 Alur Penambahan atau Pemasokan Stok

Dalam basis data berikut, setelah dilakukannya beberapa transaksi dan diketahui bahwa stok produk yang dimiliki oleh toko telah kurang dari 20 produk, maka akan dilakukan penambahan stok pada produk tersebut agar kembali menjadi stok produk awal yaitu sejumlah 25.

2.6 Output Basis Data

2.6.1 Output Pengurangan dan Penambahan Stok Menggunakan Trigger

Setelah dilakukan pengolahan basis data menggunakan aplikasi pemrograman MySQL, maka terjadi pengurangan stok pada beberapa produk yang dibeli oleh pelanggan menggunakan perintah Trigger yang dapat diperiksa pada dokumen

berekstensi sql yang terlampir. Untuk hasil atau *output* dari perintah tersebut, dapat ditunjukkan melalui sampel tangkapan layar berikut.

	ID_Produk	Kategori_Produk	Merk_Produk	Spesifikasi_Produk	Warna_Produk	nama_produk	Stok_Produk	Harga_Produk
▶	BLOE1007	Blender	Oxone	1L	Light Blue	Oxone 1L Blender Light Blue	21	1250000
	VCSG0311	Vacuum Cleaner	Samsung	Stick	Green Sage	Samsung Stick Vacuum Cleaner G...	21	2100000
	ACDN1501	AC	Daikin	Split 1,5 PK	Ivory White	Daikin Split 1,5 PK AC Ivory White	23	9500000
	ACLG1501	AC	LG	Split 1,5 PK	Ivory White	LG Split 1,5 PK AC Ivory White	23	8500000
	BLMO10007	Blender	Miyako	1L	Light Blue	Miyako 1L Blender Light Blue	23	1150000
	BLMO2007	Blender	Miyako	2L	Light Blue	Miyako 2L Blender Light Blue	23	2350000
	BLPS2006	Blender	Philips	2L	Light Green	Philips 2L Blender Light Green	23	1750000
	BLPS2007	Blender	Philips	2L	Light Blue	Philips 2L Blender Light Blue	23	1750000
	BLSP2006	Blender	Sharp	2L	Light Green	Sharp 2L Blender Light Green	23	1600000
	LEPN0109	Lemari Es	Polytron	1-Door	Bloody Red	Polytron 1-Door Lemari Es Black ...	23	1750000
	LESP0309	Lemari Es	Sharp	Multi Door	Bloody Red	Sharp Multi Door Lemari Es Blood...	23	13250000

Sampel tangkapan layar tersebut berasal dari tabel Produk yang telah berubah untuk stok produknya karena telah dilakukan pembelian oleh pelanggan. Selanjutnya, terkait penambahan stok produk, diketahui berdasarkan *records* pada tabel Detailtransaksi yang dapat diperiksa pada dokumen berekstensi sql yang terlampir terdapat salah satu produk yaitu produk dengan ID Produk VCSG0110 yang terjual 6 buah yang berakibat pada jumlah stok produk tersebut menjadi 19 buah. Maka berdasarkan trigger penambahan stok, stok produk yang kurang dari 20 produk akan kembali pada nilai 25 sesuai dengan stok awal produk. Berikut adalah sampel tangkapan layar dari tabel Produk untuk ID Produk VCSG0110 yang telah ditambahkan kembali stoknya menjadi 25.

	ID_Produk	Kategori_Produk	Merk_Produk	Spesifikasi_Produk	Warna_Produk	nama_produk	Stok_Produk	Harga_Produk
▶	VCSG0110	Vacuum Cleaner	Samsung	Upright	Orange Sunset	Samsung Upright Vacuum Clean...	25	400000
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

2.6.2 Output Procedure Penghitungan Subtotal Biaya Per Transaksi

Dalam basis data berikut, dilakukan procedure sub-total biaya per transaksi yang disebabkan karena adanya pelanggan yang membeli produk lebih dari 1 kategori/merk/spesifikasi/warna yang membuat terdapat lebih dari 1 baris yang berisi ID Transaksi yang sama namun produk yang berbeda, sehingga masing-masing harus dihitung terlebih dahulu total biaya per barang dan jumlah barang yang dibeli. Dalam perhitungan tersebut, kemudian dihitung juga *membership fee* per pembelian produk yang disesuaikan dengan jenis *membership* yang dimiliki oleh pelanggan. Kemudian dihitung juga diskon produk apabila pembelian produk dengan harga di atas 10 juta sesuai dengan jenis *membership* yang dimiliki oleh pelanggan. Kemudian dihitung juga biaya pengiriman untuk produk yang dibeli sesuai dengan jasa dan jenis pengiriman yang dipilih oleh pelanggan. Lalu dihitung juga bunga yang harus dibayarkan oleh pelanggan sesuai dengan metode pembayaran dan jenis *membership* yang dimiliki oleh pelanggan. Untuk rincian terkait data-data tersebut dapat diperiksa melalui dokumen yang terlampir. Berikut adalah sampel tangkapan layar untuk output procedure berikut.

	id_transaksi	id_kasir	customer_id	product_id	product_price	total_quantity	membership	payment_method	discount	total_bunga	shipping	total_shipping_cost	grand_total
▶	WTS-100923-0819	CAS100897	EK8696	LEPN0109	1750000.00	1	Bronze	Cash	0.000	0.00	JNE	175200.00	1932700.00
	WTS-100923-0819	CAS100897	EK8696	ACLG1501	8500000.00	2	Bronze	Cash	0.025	0.00	JNE	175200.00	16757700.00
	WTS-131023-1706	CAS100897	LW6014	LETA0108	3000000.00	1	Bronze	Cash	0.000	0.00	Self Pick-up	0.00	3007500.00
	WTS-051123-0719	CRR170907	EK8696	BLPS1006	1350000.00	1	Bronze	Cash	0.000	0.00	J&T	118260.00	1475760.00
	WTS-051123-0719	CRR170907	EK8696	LETA0309	10250000.00	1	Bronze	Cash	0.025	0.00	J&T	118260.00	10119510.00
	WTS-150423-0123	CAS100897	J87493	VCSG0410	5050000.00	1	Bronze	Credit	0.000	0.05	JNE	59000.00	5369000.00

2.6.3 Output Procedure Nota Keseluruhan ID Transaksi

Dalam MySQL yang telah dibuat, dimunculkan informasi mengenai detail transaksi yang telah dilakukan oleh *customer*. Untuk memunculkan data tersebut, memerlukan procedure yang diberi nama *generate_invoice* yang mana pada outputnya satu ID transaksi hanya muncul dalam satu baris saja. Berikut adalah sampel hasil output dari procedure yang telah dibuat.

id_transaksi	tanggal_transaksi	waktu_transaksi	customer_id	Nama_Kasir	membership	nama_produk	jumlah_produk	selling_price	mem
WTS-100923-0819	2023-09-10	08:19:18	EK8696	Adje Susanto	Bronze	AC LG Split 1,5 PK Ivory White,Lemari Es Polyt...	1, 2	18750000	7500
WTS-131023-1706	2023-10-13	17:06:48	LW6014	Adje Susanto	Bronze	Lemari Es Toshiba 1-Door Black Doff	1	3000000	7500

Pada tangkapan layar hasil procedure *generate_invoice* tersebut belum memuat semua kolom, masih ada informasi mengenai *membership fee*, *payment method*, *discount*, bunga, *shipping*, *shipping cost*, dan juga *total price* yang tidak bisa dimunculkan pada tangkapan layar diatas. Untuk hasil *output* lengkap bisa di akses pada MySQL dengan nama *procedure generate_invoice*.

2.6.4 Output Procedure Nota Per ID Transaksi

Setelah pelanggan melakukan transaksi maka data tersebut tersimpan dalam basis data berikut. Seluruh data yang berhubungan oleh suatu ID_Transaksi dapat dimunculkan melalui *procedure generate_invoice_pertransaction*, berikut merupakan hasil output dari *procedure* tersebut dengan menggunakan ID_Transaksi 'WTS-131222-1503'.

id_transaksi	tanggal_transaksi	waktu_transaksi	customer_id	Nama_Kasir	membership	nama_produk	jumlah_produk	selling_price
WTS-131222-1503	2022-12-13	15:03:43	BB8015	Andreas Hadi	Non-Member	AC Daikin Split 1,5 PK Ivory White,Vacuum Cleaner Samsung Stick Green Sage	1, 2	21100000

Table tersebut belum memuat data secara lengkap, masih terdapat kolom *selling_price*, *membership_fee*, *payment_method*, *discount*, bunga, *shipping*, *shipping_cost*, dan *total_price*. Untuk hasil *output* lengkap bisa di akses pada MySQL dengan nama *procedure generate_invoice_pertransaction*('ID_Transaksi' yang diinginkan).

2.6.5 Output Seluruh Produk Terjual

Setelah terjadinya transaksi, maka dapat dilihat berapa banyak barang yang laku terjual untuk tiap produknya. Untuk mencari berapa yang terjual untuk tiap produknya, digunakan perintah yang telah terlampir di MySQL. Untuk hasil atau output yang dimunculkan, dapat dilihat pada sampel tangkapan layar berikut.

ID_Produk	Nama_Produk	Jumlah_Produk
VCSG0511	Vacuum Cleaner Samsung Robot ...	1
VCSG0510	Vacuum Cleaner Samsung Robot ...	2

Pada tangkapan layar tersebut, dapat dilihat mengenai jumlah penjualan dari suatu produk berdasarkan kegiatan transaksi yang telah dilakukan oleh pelanggan.

2.6.6 Output Produk yang Terjual Per Bulannya

Dalam basis data berikut, setelah terjadinya transaksi maka dapat diidentifikasi terkait produk yang terjual per bulannya. Terkait perintah yang digunakan untuk memunculkan output tersebut dapat diperiksa melalui dokumen berekstensi sql yang terlampir. Berikut adalah sampel tangkapan layar dari output tersebut.

	Bulan	ID_Produk	Nama_Produk	Jumlah_Produk
▶	1	VCSG0311	Vacuum Cleaner Samsung Stick Green Sage	2
	2	BLPS2007	Blender Philips 2L Light Blue	2
	2	VCSG0510	Vacuum Cleaner Samsung Robot Orange Sunset	2
	3	BLMO2007	Blender Miyako 2L Light Blue	2
	3	LETA0408	Lemari Es Toshiba Side-by-Side Black Doff	1

Pada tangkapan layar tersebut, dapat dilihat mengenai produk yang terjual dan jumlah penjualan dari produk tersebut sesuai bulan berdasarkan kegiatan transaksi yang telah dilakukan oleh pelanggan.

2.6.7 Output Produk yang Terjual Per Bulannya *By Request*

Pada dasarnya, output ini hampir sama dengan output sebelumnya, akan tetapi untuk output ini hanya memunculkan bulan yang diinginkan saja. Terkait syntax yang berisi perintah untuk mengeluarkan output tersebut dapat diperiksa melalui dokumen berekstensi sql yang terlampir. Berikut adalah sampel tangkapan layar dari output tersebut.

	Bulan	ID_Produk	Nama_Produk	Jumlah_Produk
▶	11	BLPS1006	Blender Philips 1L Light Green	1
	11	LELG0308	Lemari Es LG Multi Door Black Doff	1
	11	LETA0309	Lemari Es Toshiba Multi Door Bloody Red	1
	11	LETA0409	Lemari Es Toshiba Side-by-Side Bloody Red	2
	11	RCMO1005	Rice Cooker Miyako 1L Caramel Brown	1
	11	VCBH0311	Vacuum Cleaner Bosch Stick Green Sage	2
	11	VCPC0111	Vacuum Cleaner Panasonic Upright Green Sage	2
	11	VCSG0110	Vacuum Cleaner Samsung Upright Orange Sunset	2
	11	VCSG0410	Vacuum Cleaner Samsung Handheld Orange Su...	1

2.6.8 Output Produk yang Paling Laku

Pada program MySQL berikut, dibuat perintah untuk memunculkan produk dengan penjualan terbanyak (paling laku). Dengan output yang dimunculkan adalah ID produk, kategori, merk, spesifikasi, warna, harga, dan banyak penjualan dari produk tersebut. Output yang muncul didapat dari perintah *order by* total pembelian dengan *limit* 1. Berikut adalah tangkapan layar hasil output perintah yang telah dijalankan.

	ID_Produk	Kategori_Produk	Merk_Produk	Spesifikasi_Produk	Warna_Produk	Harga_Produk	Total_Pembelian
▶	VCSG0110	Vacuum Cleaner	Samsung	Upright	Orange Sunset	400000	6

Pada tangkapan layar tersebut, dapat dilihat hasil output yang memunculkan produk dengan penjualan tertinggi dan juga jumlah penjualan produk tersebut.

2.6.9 Output Produk yang Kurang Laku

Dalam basis data MySQL, telah dirancang perintah untuk memunculkan produk dengan penjualan sedikit (satu penjualan). Untuk memunculkan produk dengan penjualan minimum, perlu melakukan perintah *select*, *from*, *join*, *groupby*, dan *having* seperti pada dokumen MySQL terlampir. Selain memunculkan pembelian minimum, bisa memunculkan produk dengan total pembelian yang diinginkan dengan mengganti angka pada *having* total pembelian. Berikut adalah tangkapan layar output dari perintah untuk memunculkan produk dengan penjualan minimum.

	ID_Produk	Kategori_Produk	Merk_Produk	Spesifikasi_Produk	Warna_Produk	Harga_Produk	Total_Pembelian
▶	VCPC0110	Vacuum Cleaner	Panasonic	Upright	Orange Sunset	600000	1
	LESP0208	Lemari Es	Sharp	2-Door	Black Doff	4750000	1

BAB III

PENUTUP

Kesimpulan

Implementasi basis data pada "Wisantoko Store's" membawa dampak positif yang signifikan dalam berbagai aspek operasional dan kinerja toko. Manfaat utamanya terlihat pada manajemen stok yang lebih efisien, memungkinkan toko untuk mengelola inventaris dengan akurat dan menghindari kekurangan atau kelebihan stok. Hal ini terbukti melalui data penjualan produk, dengan Vacuum Cleaner Samsung Upright Orange Sunset menjadi produk paling laris sebanyak 6 buah, sementara jenis produk lemari es kurang diminati dengan penjualan sebanyak 11 buah dengan tiap spesifikasi yang berbeda.

Implementasi basis data juga mencakup informasi tentang jumlah produk yang laku di pasaran sejak awal transaksi, bersama dengan data nama dan jumlah produk yang terjual setiap bulannya. Dengan data ini, pemilik toko dapat mengidentifikasi produk yang perlu diprioritaskan dalam menjaga stok agar tetap memenuhi kebutuhan pelanggan.

Basis data ini juga mengintegrasikan sistem otomatis yang melakukan transaksi dengan supplier ketika stok barang mencapai kurang dari 20 buah, menjaga agar jumlahnya kembali ke 25 buah. Sistem ini diatur oleh suatu "trigger" yang mengurangi stok otomatis setiap kali terjadi transaksi. Pemilik toko juga memiliki akses ke data detail transaksi berdasarkan ID_Transaksi yang tercetak pada nota, memungkinkan pemantauan transaksi secara lebih efektif dan akurat. Dengan demikian, implementasi basis data tidak hanya meningkatkan efisiensi operasional, tetapi juga memberikan alat yang kuat bagi pemilik toko untuk mengelola stok dan transaksi dengan lebih efektif.

Saran

Untuk meningkatkan manfaat implementasi basis data pada "Wisantoko Store's," disarankan untuk memperluas fungsionalitas sistem dengan beberapa fitur tambahan yang dapat memberikan informasi lebih komprehensif. Pertama, dapat ditambahkan fitur pencatatan data setiap *order supply* yang dilakukan ke *supplier*, mencakup detail produk, jumlah, harga, dan tanggal transaksi. Hal ini akan membantu pemilik toko untuk melacak dengan lebih teliti aliran stok barang dan mengoptimalkan proses pengadaan. Selain itu, disarankan untuk menyertakan fitur pencatatan data keuangan, termasuk dana masuk dan keluar dari toko, seperti pendapatan dari penjualan, biaya operasional, dan pembayaran ke supplier. Fungsi ini akan memberikan gambaran yang lebih lengkap mengenai kondisi keuangan toko dan memfasilitasi perencanaan keuangan yang lebih efektif. Terakhir, direkomendasikan untuk mengintegrasikan perhitungan total keuntungan toko ke dalam sistem. Dengan mempertimbangkan seluruh pendapatan dari penjualan dan mengurangnya dengan biaya operasional serta pengeluaran lainnya, fitur ini akan memberikan gambaran yang jelas mengenai kesehatan keuangan toko dan dapat digunakan sebagai dasar untuk pengambilan keputusan. Penerapan fitur-fitur ini akan memberikan pemilik toko akses yang lebih baik terhadap data operasional dan keuangan, memudahkan pengelolaan stok, pengadaan, dan keuangan toko secara lebih terintegrasi dan efisien.

LAMPIRAN

<https://its.id/m/RawDataWisantokoStore>