

**LAPORAN TUGAS BESAR BASIS DATA
DESAIN BASIS DATA
UNTUK PLATFORM PENJUALAN PRODUK UMKM**

Laporan ini disusun untuk memenuhi Tugas Mata Kuliah Basis Data Praktikum.



Disusun oleh:

Fitra Rifki Firdaus	231524008
Fredy Kurniadi	231524010
Zahra Hilyatul Jannah	231524031

**PROGRAM STUDI D4 TEKNIK INFORMATIKA
JURUSAN TEKNIK KOMPUTER DAN INFORMATIKA
POLITEKNIK NEGERI BANDUNG**

2024

DAFTAR ISI

DAFTAR ISI.....	1
DAFTAR GAMBAR	2
DAFTAR TABEL.....	3
1. PENDAHULUAN	4
1.1 Latar Belakang	4
1.3 Batasan Masalah.....	5
2. ANALISIS	5
2.1 Overview Sistem	5
2.2 Event.....	0
2.3 Model Data ER.....	0
2.3.1 Logical Data Model.....	0
2.3.2 Relational (Physical) Data Model.....	1
3. IMPLEMENTASI PENGOLAHAN DATA	1
3.1 DDL	1
3.2 View	2
3.3 DML	2
4. KESIMPULAN.....	0
5. LESSON LEARNED	0
DAFTAR PUSTAKA	0

DAFTAR GAMBAR

Gambar 1 Context Diagram.....	5
-------------------------------	---

DAFTAR TABEL

Tabel 1 Event Menambahkan Produk.....	Error! Bookmark not defined.
Tabel 2 Event: Meng-update Stok	Error! Bookmark not defined.
Tabel 3 Event: Membuat Akun.....	Error! Bookmark not defined.
Tabel 4 Event: Pembeli Menambahkan Barang ke Wishlist....	Error! Bookmark not defined.
Tabel 5 Event: Pembeli Melakukan Pembayaran	Error! Bookmark not defined.

1. PENDAHULUAN

1.1 Latar Belakang

Dalam era digital saat ini, banyak pengusaha dan pelaku usaha kecil yang ingin memasarkan produk mereka melalui platform online. Kementerian Koperasi dan UKM mencatat bahwa hingga Februari 2022, sebanyak 17,25 juta pelaku UMKM sudah terhubung ke dalam ekosistem digital. Namun, tingginya biaya yang dikenakan oleh marketplace konvensional, seperti biaya administrasi dan biaya program promosi tambahan, sering kali membatasi potensi keuntungan yang dapat diperoleh oleh para pengusaha. Hal ini mendorong mereka untuk mencari alternatif, seperti memiliki platform sendiri untuk mendukung proses penjualan dan manajemen data.

Pembuatan model database yang baik dan terstruktur dapat membantu pengusaha dalam mengelola informasi produk, transaksi, dan data pengguna dengan lebih efektif. Database ini dapat digunakan sebagai fondasi sistem penjualan online yang lebih efisien, yang memfasilitasi pengelolaan data tanpa ketergantungan pada marketplace eksternal.

Dari sudut pandang konsumen, ekosistem digital yang semakin berkembang juga memengaruhi ekspektasi mereka terhadap pengalaman belanja online. Kemudahan akses, transparansi harga, dan personalisasi layanan menjadi faktor penting dalam menentukan loyalitas konsumen. Namun, banyak konsumen merasa terbebani oleh harga produk yang meningkat akibat biaya tambahan dari marketplace konvensional, seperti biaya administrasi atau program promosi yang diteruskan kepada mereka oleh penjual.

Selain itu, konsumen juga mengkhawatirkan keamanan data pribadi mereka, terutama ketika platform marketplace besar mengalami kebocoran data. Oleh karena itu, mereka mulai mengapresiasi platform independen yang menawarkan harga lebih kompetitif, transparansi dalam transaksi, dan jaminan privasi data yang lebih baik.

Dengan adanya platform penjualan yang dikelola langsung oleh pelaku usaha, konsumen dapat merasakan manfaat seperti harga yang lebih rendah, hubungan yang lebih langsung dengan penjual, serta pengalaman berbelanja yang lebih personal dan aman. Hal ini menjadi pendorong penting bagi para pelaku usaha untuk membangun sistem yang tidak hanya mendukung bisnis mereka tetapi juga memenuhi kebutuhan dan ekspektasi konsumen secara efektif.

[Kemenkop UKM: Sudah 17,25 Juta UMKM yang Terhubung ke Platform Digital](#)

1.2 Definisi Masalah

Berdasarkan latar belakang yang telah dijelaskan, permasalahan yang perlu dipecahkan adalah bagaimana membangun model database yang dapat memfasilitasi interaksi antara pembeli dan penjual, serta mengelola data dengan baik. Model database ini harus mencakup entitas dan relasi yang dapat mendukung fitur-fitur sebagai berikut:

- **Pembeli:** Meliputi penyimpanan data untuk registrasi, login, pengelolaan wishlist, pemesanan barang, dan riwayat pembelian.

- **Penjual:** Meliputi penyimpanan data untuk registrasi, login, pengelolaan produk, menerima dan memproses pesanan, serta manajemen data toko.

Model database akan mengimplementasikan konsep CRUD (Create, Read, Update, Delete) pada setiap entitas yang relevan untuk memastikan pengguna dapat mengelola data mereka dengan mudah.

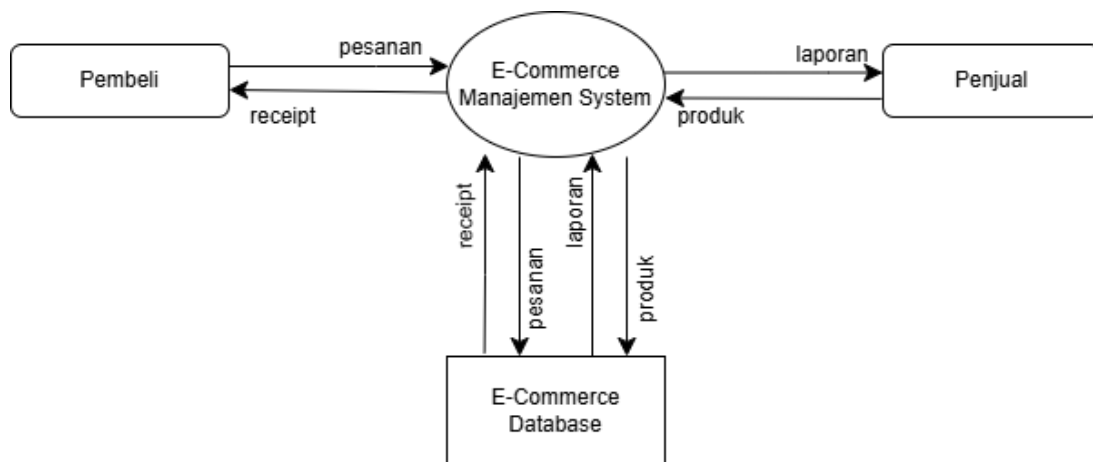
1.3 Batasan Masalah

Batasan dalam pengembangan model database ini adalah sebagai berikut:

- Fokus utama adalah pada pengelolaan data pembeli dan penjual dalam bentuk model database.
- Model database hanya akan mencakup skema data dan relasi untuk transaksi pembelian barang, tanpa mencakup aspek pengiriman barang.
- Database akan dirancang menggunakan sistem manajemen basis data relasional (RDBMS) dengan fitur yang memungkinkan operasi CRUD pada data pembeli, penjual, produk, dan transaksi.

2. ANALISIS

2.1 Overview Sistem



Gambar 1 Context Diagram

Sistem yang dirancang merupakan *E-commerce Management System* yang berfungsi sebagai perantara antara pengguna—baik pembeli maupun penjual—dan *database E-Commerce* yang mengelola data transaksi dan inventori produk. Pembeli dapat melakukan berbagai aktivitas seperti melakukan pemesanan produk yang tersedia di sistem. Sistem ini akan menerima pesanan pembeli, memprosesnya, dan mengirimkan data transaksi dalam bentuk *receipt* yang tersimpan di *database E-Commerce* untuk keperluan pelacakan riwayat transaksi dan pemenuhan pesanan.

Di sisi lain, penjual dapat menggunakan *E-commerce Management System* untuk menambahkan, memperbarui, dan menghapus produk yang dijual. Setiap perubahan data produk yang dilakukan oleh penjual akan tercatat dalam *database E-Commerce* dan digunakan sebagai referensi untuk laporan inventori dan ketersediaan barang. Sistem ini juga menghasilkan laporan yang dapat diakses oleh penjual untuk memantau performa penjualan dan memastikan kelancaran manajemen stok produk. Dengan demikian, *E-commerce Management System* ini memfasilitasi interaksi dan sinkronisasi data secara efisien antara pembeli, penjual, dan *database E-Commerce*, mendukung keseluruhan proses penjualan secara digital.

2.2 Event

Berikut adalah beberapa event dan respon sistem yang terjadi dalam aplikasi:

- 1. Registrasi Akun - Mendaftarkan akun baru untuk mengakses fitur e-commerce.

Deskripsi				
Respon Sistem				
No	Deskripsi Respon	Input	Output	Aturan
1				
2				
3				
4				
5				

Kamus Data:

-

CATATAN: lakukan analisis kebutuhan indexing

Nama					
No.	Nama Kolom	Tipe Data	Constraint	Deskripsi	Contoh Data
1.					
2.					
3.					
4.					
5.					
6.					

2. Login ke Akun - Masuk ke akun untuk mengakses profil dan riwayat pesanan.
3. Menambahkan Alamat Baru - Menambahkan alamat pengiriman baru untuk transaksi.
4. Memperbarui Alamat Pengiriman - Mengedit atau memperbarui alamat yang sudah ada.
5. Memperbarui Data Diri - Mengedit atau memperbarui informasi pribadi dalam profil.
6. Menambahkan Barang ke Wishlist - Menyimpan produk favorit ke dalam wishlist untuk pembelian di masa mendatang.
7. Menghapus Barang dari Wishlist - Menghapus barang dari wishlist jika sudah tidak diinginkan.
8. Menambahkan Produk Baru - Menambah produk baru ke dalam sistem e-commerce.

Tabel 1 Event Menambahkan Produk

Deskripsi

Penjual menambahkan produk baru ke dalam database dengan nama, kategori, dan deskripsi yang sesuai. Produk memiliki ID unik yang dihasilkan oleh sistem.

Respon Sistem

No	Deskripsi Respon	Input	Output	Aturan
1	Penjual menginputkan detail produk baru, termasuk nama, kategori, dan deskripsi.	Nama_Produk, Kategori, Deskripsi	ID_Produk, pesan berhasil	Nama produk tidak boleh sama dengan yang sudah ada.
2	Sistem memvalidasi kategori dan nama produk.	Nama_Produk, Kategori	Validasi berhasil/gagal	Kategori harus termasuk dalam daftar yang ditentukan. Jika nama produk sudah ada, tampilkan pesan kesalahan.
3	Sistem menghasilkan ID produk baru berdasarkan format: PROD + kode kategori + nomor urut (2 digit)	Kategori	ID_Produk	ID produk harus unik. Format ID: PROD + tiga huruf kode kategori + nomor urut (contoh: PRODTOP01, PRODBOT02).

Kamus Data:

- **Produk:** {ID_Produk, Nama_Produk, Deskripsi, Harga, Kategori, Stok}

CATATAN: lakukan analisis kebutuhan indexing

Nama		Produk			
No.	Nama Kolom	Tipe Data	Constraint	Deskripsi	Contoh Data
1.	ID_Produk	VARCHAR	PK	Identifier unik (Primary Key) dari tabel produk, terdiri dari format PROD + kode kategori + urutan	PRODTOP01
2.	Nama_Produk	VARCHAR	UNIQUE, NOT NULL		T-shirt polos putih
3.	Deskripsi	CLOB	NOT NULL	Deskripsi lengkap tentang produk	T-shirt polos putih ini terbuat dari bahan katun berkualitas tinggi yang nyaman dipakai sepanjang hari. Desainnya sederhana dan cocok digunakan untuk gaya kasual atau sebagai kaos dasar untuk dipadukan dengan pakaian lain. Tersedia dalam berbagai ukuran, T-shirt ini memiliki potongan yang pas di badan dan jahitan yang rapi, memberikan kesan stylish namun tetap nyaman. Ideal untuk dipakai sehari-hari atau sebagai outfit santai.

9. Menambahkan Detail Produk Baru

Deskripsi				
<p>Penjual menambahkan detail produk baru ke dalam database. Detail produk mencakup ukuran, stok, harga reguler, dan diskon (opsional). Setiap detail produk memiliki ID unik yang dihasilkan sistem dengan format:</p> <p>D + ID_Produk + nomor urut (2 digit).</p>				
Respon Sistem				
No	Deskripsi Respon	Input	Output	Aturan
1	Penjual menambahkan detail produk baru, termasuk ukuran, stok, harga, dan diskon.	ID_Produk, Ukuran, Stok, Harga_Reguler, Diskon (opsional)	ID_Detail_Produk, pesan berhasil	Ukuran tidak boleh duplikat untuk ID produk yang sama.
2	Sistem memvalidasi apakah kombinasi ID_Produk dan Ukuran sudah ada.	ID_Produk, Ukuran	Validasi berhasil/gagal	Jika kombinasi sudah ada, tampilkan pesan kesalahan dan proses dihentikan.

3	Sistem mencari nomor urut terakhir berdasarkan ID_Produk yang diberikan.	ID_Produk	Nomor urut	Nomor urut baru = nomor urut terakhir + 1.
4	Sistem membuat ID_Detail_Produk baru dengan format: 'D' + ID_Produk + nomor urut.	ID_Produk, Nomor urut	ID_Detail_Produk	Format ID: 'D' + ID_Produk + 2 digit nomor urut.
5	Sistem menyimpan detail produk baru ke dalam tabel detailproduk.	ID_Detail_Produk, Ukuran, Stok, Harga_Reguler, ID_Produk, Diskon	Pesan berhasil	Jika operasi berhasil, data disimpan dan transaksi dikomit. Jika gagal, rollback dan tampilkan pesan kesalahan.

Kamus Data:

- Detail Produk { ID_Detail_Produk, Ukuran, Stok, Harga_Reguler }

CATATAN: lakukan analisis kebutuhan indexing

Nama		detailproduk			
No.	Nama Kolom	Tipe Data	Constraint	Deskripsi	Contoh Data
1.	ID_Detail_Produk	VARCHAR	PK	Identifier unik untuk detail produk, terdiri dari format: 'D' + ID_Produk + nomor urut (2 digit).	DPRODTOP0101

2.	Ukuran	VARCHAR	UNIQUE, NOT NULL	Ukuran produk, misalnya S, M, L, XL.	M
3.	Stok	INTEGER	NOT NULL	Jumlah stok yang tersedia untuk ukuran tertentu.	50
4.	Harga_Reguler	NUMBER(10,2)	NOT NULL	Harga reguler untuk detail produk tertentu.	120000

10. Mengedit Produk

Deskripsi

Prosedur ini digunakan untuk mengedit informasi sebuah produk berdasarkan ID_Produk yang diberikan. Informasi yang dapat diubah mencakup nama produk, kategori, dan deskripsi. Prosedur ini juga memvalidasi kategori yang diberikan agar sesuai dengan daftar kategori yang diizinkan.

Respon Sistem

No	Deskripsi Respon	Input	Output	Aturan
1	Penjual memasukkan ID produk beserta data baru: nama, kategori, dan deskripsi.	ID_Produk, Nama_Produk, Kategori,	Validasi berhasil/gagal	ID_Produk harus valid (ada di tabel produk). Jika tidak ditemukan, tampilkan pesan

		Deskripsi		kesalahan.
2	Sistem memvalidasi kategori yang diberikan.	Kategori	Kategori valid/gagal	Kategori harus termasuk dalam daftar: tops, cardigan and jacket, dress, skirts, accessoris.
3	Sistem memperbarui data produk berdasarkan ID_Produk yang valid.	ID_Produk, Nama_Produk, Kategori, Deskripsi	Pesan berhasil/gagal	Jika ID produk valid, data akan diperbarui. Jika tidak ditemukan, tampilkan pesan kesalahan.
4	Sistem menyimpan perubahan ke database.	Data produk baru	Pesan berhasil/gagal	Jika operasi berhasil, data dikomit. Jika gagal, rollback dilakukan.

Kamus Data:

- {ID_Produk, Stok, Harga_Reguler, Id_diskon}

CATATAN: lakukan analisis kebutuhan indexing

Nama		Produk			
No.	Nama Kolom	Tipe Data	Constraint	Deskripsi	Contoh Data

1.	ID_Produk	VARCHAR	PK	Identifier unik (Primary Key) dari tabel produk, terdiri dari format PROD + kode kategori + urutan	PRODTOP01
2.	Nama_Produk	VARCHAR	UNIQUE, NOT NULL	Nama produk yang dijual	T-shirt polos putih
3.	Deskripsi	CLOB	NOT NULL	Deskripsi lengkap tentang produk	T-shirt polos putih ini terbuat dari bahan katun berkualitas tinggi yang nyaman dipakai sepanjang hari. Desainnya sederhana dan cocok digunakan untuk gaya kasual atau sebagai kaos dasar untuk dipadukan dengan pakaian lain. Tersedia dalam berbagai ukuran, T-shirt ini memiliki potongan yang pas di badan dan jahitan yang rapi, memberikan kesan stylish namun tetap nyaman. Ideal untuk dipakai sehari-hari atau sebagai outfit santai.

11. Mengedit Detail Produk - Memperbarui deskripsi, kategori, atau detail lainnya dari produk yang sudah ada.

Deskripsi

Prosedur ini digunakan untuk mengedit informasi detail sebuah produk berdasarkan ID_Detail_Produk yang diberikan. Informasi yang dapat

diubah mencakup stok, harga reguler, dan ID diskon (opsional). Sistem memvalidasi ID diskon jika diberikan dan memastikan ID detail produk yang dimasukkan valid.

Respon Sistem

No	Deskripsi Respon	Input	Output	Aturan
1	Penjual memasukkan ID detail produk dan data baru: stok, harga reguler, ID diskon.	ID_Detail_Produk, Stok, Harga_Reguler, ID_Diskon	Validasi berhasil/gagal	ID_Detail_Produk harus valid (ada di tabel detailproduk). Jika tidak valid, tampilkan pesan kesalahan.
2	Sistem memvalidasi keberadaan ID_Detail_Produk dan mengambil data terkait.	ID_Detail_Produk	Detail ditemukan/tidak	Jika ID_Detail_Produk tidak ditemukan, prosedur dihentikan dan pesan kesalahan ditampilkan.
3	Sistem memvalidasi ID diskon jika diberikan.	ID_Diskon	ID diskon valid/gagal	Jika ID diskon tidak valid (tidak ada di tabel diskon), prosedur dihentikan.

4	Sistem memperbarui data stok, harga reguler, dan ID diskon pada tabel detail_produk.	ID_Detail_Produk, Stok, Harga_Reguler, ID_Diskon	Pesan berhasil/gagal	Jika validasi berhasil, data diperbarui. Jika tidak, rollback dilakukan.
---	--	---	-------------------------	--

Kamus Data:

- Detail Produk {ID_Detail_Produk, Ukuran, Stok, Harga_Reguler, ID_Produk, ID_Diskon}

CATATAN: lakukan analisis kebutuhan indexing

Nama		detailproduk			
No.	Nama Kolom	Tipe Data	Constraint	Deskripsi	Contoh Data
1.	ID_Detail_Produk	VARCHAR	PK	Identifier unik untuk detail produk, terdiri dari format: 'D' + ID_Produk + nomor urut (2 digit).	DPRODTOP0101
3.	Stok	INTEGER	NOT NULL	Jumlah stok yang tersedia untuk ukuran tertentu.	50
4.	Harga_Reguler	NUMBER(10,2)	NOT NULL	Harga reguler untuk detail produk tertentu.	120000
5	Id_diskon	VARCHAR		Identifier unik untuk diskon.	DISK001

12. Memperbarui Harga Produk - Mengubah harga produk sesuai kebutuhan.

Deskripsi				
Pemilik usaha melakukan pembaruan pada harga reguler dari suatu detail produk yang ada dalam database				
Respon Sistem				
No	Deskripsi Respon	Input	Output	Aturan
1	Penjual menginputkan ID detail produk dan harga baru.	ID_Detail_Produk, Harga_Baru	Pesan berhasil/gagal	ID detail produk harus sudah ada di tabel detailproduk.
2	Sistem memvalidasi keberadaan ID detail produk.	ID_Detail_Produk	Validasi berhasil/gagal	Jika ID tidak ditemukan, tampilkan pesan kesalahan.
3	Sistem memvalidasi bahwa harga baru lebih besar atau sama dengan 0.	Harga_Baru	Validasi berhasil/gagal	Jika harga baru kurang dari 0, tampilkan pesan kesalahan.
4	Sistem memperbarui harga reguler dari detail produk di tabel detailproduk.	ID_Detail_Produk, Harga_Baru	Data diperbarui di tabel Produk	Jika data berhasil diperbarui, tampilkan pesan bahwa harga telah berhasil diperbarui. Jika gagal, rollback.
5				
Kamus Data:				
•				

CATATAN: lakukan analisis kebutuhan indexing

Nama		detailproduk			
No.	Nama Kolom	Tipe Data	Constraint	Deskripsi	Contoh Data
1.	Id_detail_produk				
2.	Harga_baru				

13. Menambahkan Stok Produk - Memperbarui stok dengan cara menambah stok.

Deskripsi

Prosedur ini digunakan untuk menambahkan jumlah stok pada detail produk tertentu. Prosedur akan memvalidasi bahwa ID_Detail_Produk yang diberikan valid dan stok yang ditambahkan lebih besar dari 0. Setelah validasi, stok akan diperbarui dan hasilnya ditampilkan.

Respon Sistem

No	Deskripsi Respon	Input	Output	Aturan
1	Penjual memasukkan ID detail produk dan jumlah stok yang akan ditambahkan.	ID_Detail_Produk, Stok_Ditambahkan	Validasi berhasil/gagal	ID_Detail_Produk harus valid (ada di tabel detailproduk). Jika tidak valid, tampilkan pesan kesalahan.
2	Sistem memvalidasi jumlah stok yang ditambahkan.	Stok_Ditambahkan	Validasi berhasil/gagal	Stok yang ditambahkan harus lebih besar dari 0. Jika tidak,

				prosedur dihentikan dengan pesan kesalahan.
3	Sistem memperbarui stok pada tabel detailproduk.	ID_Detail_Produk, Stok_Ditambahkan	Pesan berhasil/gagal	Jika validasi berhasil, stok diperbarui dan pesan konfirmasi ditampilkan. Jika gagal, rollback dilakukan.

Kamus Data:

- Detail Produk {ID_Detail_Produk, Ukuran, Stok, Harga_Reguler, ID_Produk, ID_Diskon}

CATATAN: lakukan analisis kebutuhan indexing

Nama					
No.	Nama Kolom	Tipe Data	Constraint	Deskripsi	Contoh Data
1.					
2.					

14. Menghapus Produk - Menghapus produk dari sistem jika tidak dijual lagi.

Deskripsi

Prosedur ini digunakan untuk menghapus produk beserta detailnya dari database berdasarkan ID_Produk yang diberikan. Sistem akan memvalidasi apakah produk dengan ID tersebut ada. Jika ditemukan, semua data terkait pada tabel detailproduk akan dihapus terlebih

dahulu, lalu data pada tabel produk akan dihapus.

Respon Sistem

No	Deskripsi Respon	Input	Output	Aturan
1	Penjual memasukkan ID produk yang akan dihapus.	ID_Produk	Validasi berhasil/gagal	ID produk harus valid (ada di tabel produk). Jika tidak valid, tampilkan pesan kesalahan.
2	Sistem mengecek apakah produk dengan ID yang diberikan ada di database.	ID_Produk	Produk ditemukan/tidak	Jika produk tidak ditemukan, prosedur dihentikan dan pesan kesalahan dikembalikan.
3	Sistem menghapus semua data terkait di tabel detailproduk berdasarkan ID produk.	ID_Produk	Pesan berhasil/gagal	Semua detail terkait dengan produk akan dihapus sebelum menghapus produk di tabel utama.
4	Sistem menghapus produk dari tabel produk jika semua data terkait sudah	ID_Produk	Pesan berhasil/gagal	Produk berhasil dihapus jika tidak ada kendala, atau rollback jika terjadi

	dihapus.			kesalahan selama proses.
--	----------	--	--	--------------------------

Kamus Data:

- Detail Produk {ID_Detail_Produk, Ukuran, Stok, Harga_Reguler, ID_Produk, ID_Diskon}

CATATAN: lakukan analisis kebutuhan indexing

Nama					
No.	Nama Kolom	Tipe Data	Constraint	Deskripsi	Contoh Data
1.					
2.					

15. Menambahkan Diskon pada Produk - Menambahkan diskon pada produk tertentu.

Deskripsi

Prosedur ini digunakan untuk menambahkan data diskon ke dalam tabel diskon dan memperbarui semua detail produk yang sesuai dengan ID produk tertentu. Sistem akan memvalidasi tanggal mulai dan tanggal berakhir diskon, serta membuat ID diskon secara otomatis dengan format DSK###.

Respon Sistem

No	Deskripsi Respon	Input	Output	Aturan
1	Penjual memasukkan data diskon baru, termasuk ID produk, nama diskon, tanggal mulai, tanggal berakhir, diskon, dan maksimal diskon.	ID_Produk, Nama_Diskon, Tanggal_Mulai, Tanggal_Berakhir, Diskon, Maksimal_Diskon	Validasi berhasil/gagal	Tanggal mulai harus lebih besar dari hari ini. Tanggal berakhir harus lebih besar dari tanggal mulai.
2	Sistem membuat ID diskon baru secara otomatis.	Tidak ada	ID_Diskon	ID diskon di-generate dengan format DSK### berdasarkan data terakhir di tabel diskon.
3	Sistem menyimpan data diskon baru ke dalam tabel diskon.	ID_Diskon, Nama_Diskon, Tanggal_Mulai, Tanggal_Berakhir, Diskon, Maksimal_Diskon	Pesan berhasil/gagal	Data diskon baru ditambahkan ke tabel diskon. Jika gagal, rollback dilakukan.

4	Sistem memperbarui ID diskon pada tabel detailproduk untuk produk terkait.	ID_Produk, ID_Diskon	Pesan berhasil/gagal	Semua detail produk dengan ID produk yang sesuai diperbarui untuk menggunakan ID diskon baru.
---	--	-------------------------	-------------------------	---

Kamus Data:

- Diskon {ID_Diskon, Nama_Diskon, Tanggal_Mulai, Tanggal_Berakhir, Diskon, Maksimal_Diskon}

CATATAN: lakukan analisis kebutuhan indexing

Nama					
No.	Nama Kolom	Tipe Data	Constraint	Deskripsi	Contoh Data
1.	ID_Diskon	VARCHAR(32)	PK, NOT NULL	Identifier unik untuk diskon.	DSK001
2.	Nama_Diskon	VARCHAR(100)	NOT NULL	Nama atau deskripsi singkat untuk diskon.	Promo Akhir Tahun
3.	Tanggal_Mulai	DATE	NOT NULL	Tanggal dimulainya diskon.	2024-12-05
4.	Tanggal_Berakhir	DATE	NOT NULL	Tanggal berakhirnya diskon.	2024-12-31
5.	Diskon	FLOAT	NOT NULL	Persentase diskon yang diberikan.	20.0

16. Melihat Laporan Penjualan - Memeriksa laporan penjualan harian, mingguan, atau bulanan.

17. Menambahkan Barang ke Keranjang - Menambahkan barang ke keranjang belanja untuk persiapan pembelian.

Deskripsi

Respon Sistem				
No	Deskripsi Respon	Input	Output	Aturan
1				
2				
3				
4				
5				

Kamus Data:

-

CATATAN: lakukan analisis kebutuhan indexing

Nama	
------	--

No.	Nama Kolom	Tipe Data	Constraint	Deskripsi	Contoh Data
1.					
2.					
3.					
4.					
5.					
6.					

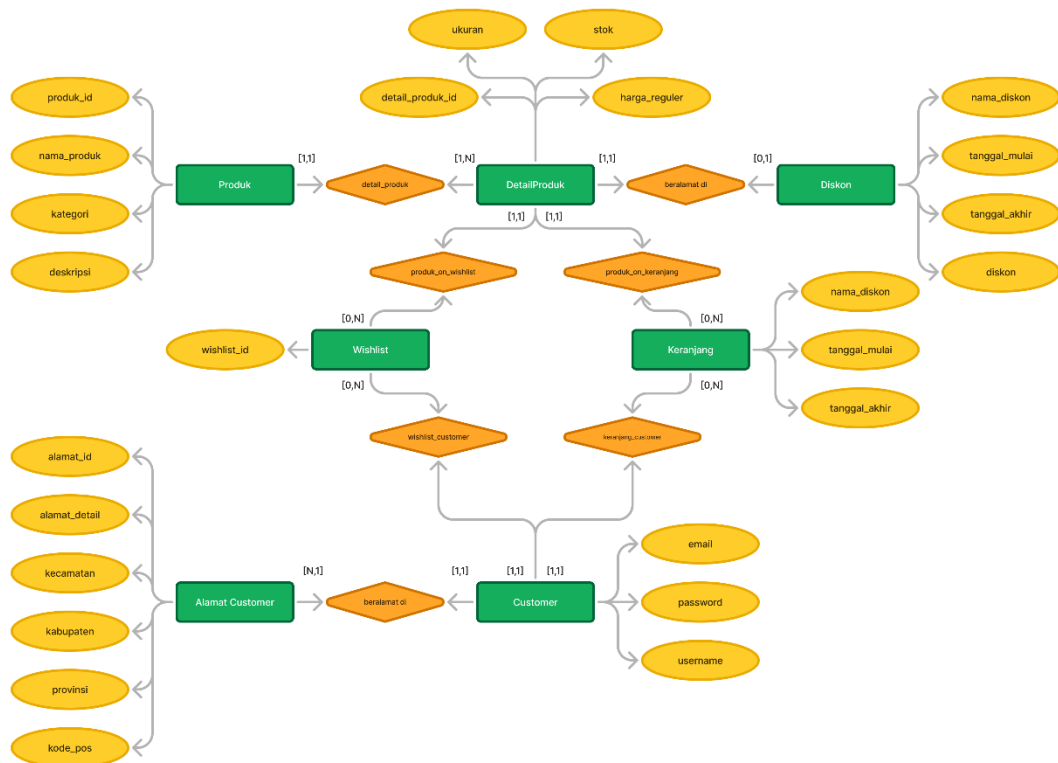
18. Menghapus Barang dari Keranjang - Menghapus barang dari keranjang jika tidak jadi dibeli.
19. Checkout (Keranjang ke Pembayaran) - Melanjutkan ke proses checkout dengan barang-barang di keranjang.
20. Memilih Metode Pembayaran - Memilih metode pembayaran yang diinginkan saat checkout.
21. Update Status Pembayaran - Memperbarui status pembayaran setelah divalidasi.
22. Melihat Riwayat Transaksi - Memeriksa dan meninjau riwayat pesanan yang telah dibuat.

2.3 Model Data ER

2.3.1 Logical Data Model

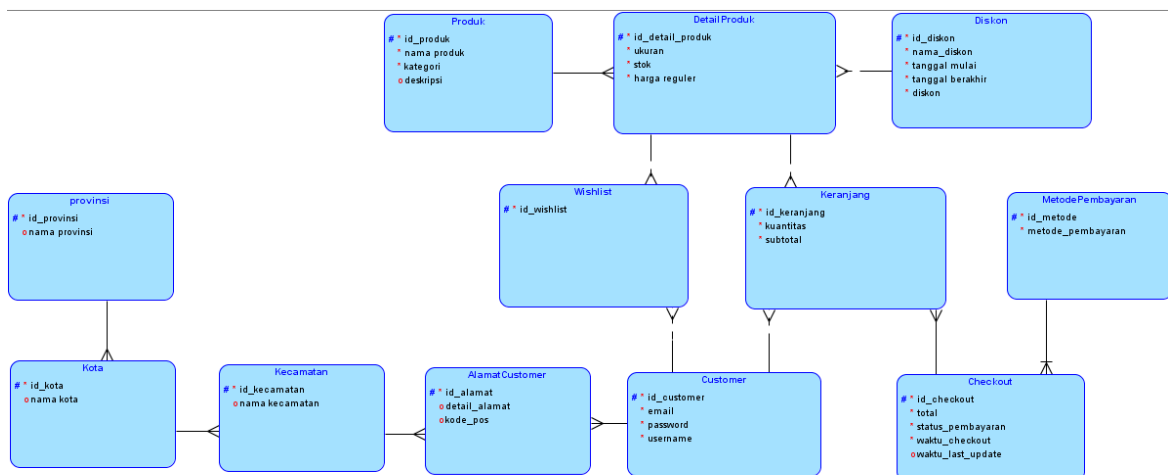
A. Notasi Chen

Pada bagian ini Anda diminta untuk membuat ERD dengan **Notasi Chen**. Tentukan nilai minimum dan maksimum dari kardinalitas!



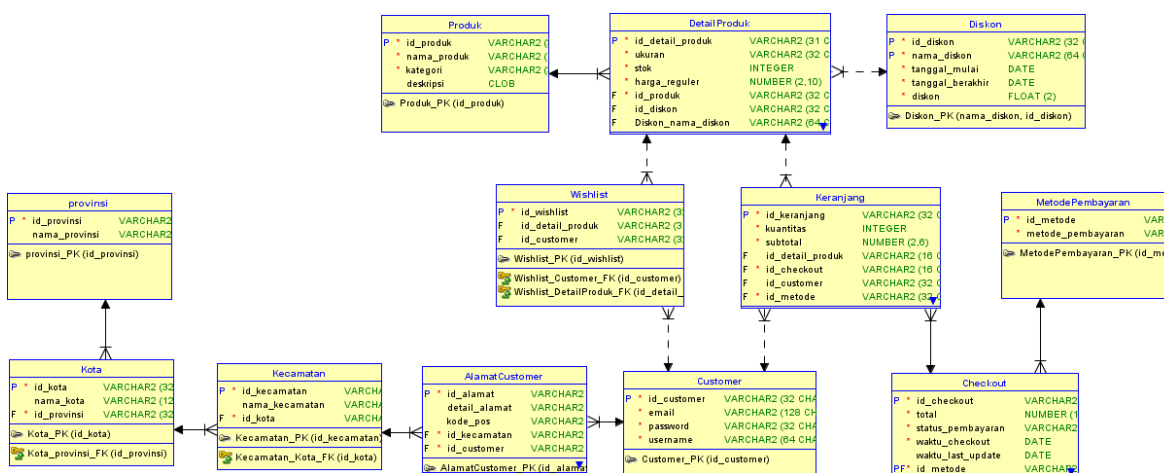
B. Oracle Data Modeller

Pada bagian ini Anda diminta untuk membuat CDM dengan **menggunakan Oracle Data Modeller (Logical)**.



2.3.2 Relational (Physical) Data Model

Pada bagian ini Anda diminta untuk membuat PDM dengan menggunakan Oracle data Modeller (Relational).



3. IMPLEMENTASI PENGOLAHAN DATA

3.1 DDL

NamaTabel	<Tabel 1>
Deskripsi	<DeskripsiTabel>
Script SQL	
<Script SQL>	

Screenshot Hasil
<Gambar Hasil Percobaan>

3.2 View

Pada bagian ini Anda diminta membuat view minimal sebanyak 2 buah berdasarkan kebutuhan reporting dari ruang lingkup yang sudah ditentukan.

Nama	customer_report	
Deskripsi	View ini ditujukan untuk memudahkan pembuatan report customer baru setiap bulannya	
No.	Nama Kolom	Tipe Data
1.	<Nama Kolom 1>	<TipeData(Length)>
2.	<Nama Kolom 2>	<TipeData(Length)>
3.	<Nama Kolom 3>	<TipeData(Length)>
Script SQL		
<Script SQL>		
Screenshot Hasil		
<Gambar Hasil Percobaan>		

3.3 DML

Nama Fitur	Registrasi Customer		
Deskripsi	<DeskripsiFitur>		
Script SQL			
INSERT INTO customer VALUES (CUSTOMER_ID_SEQ.nextval, 'Paul Jason Klein', 120000, '085364412365');			
Screenshot Hasil			
30	79 Syifa Apprila	75000	85776422447
31	80 Putri Julieta Pratiwi	69430	81219550857
32	92 Paul Jason Klein	120000	85364412365

CATATAN: Pastikan semua proses pengolahan data sudah dibuat DML sesuai ruang lingkup!

4. KESIMPULAN

Pada bagian ini Anda menuliskan kesimpulan berisi progress pengerjaan Tugas Besar (sudah atau belum) beserta kendala yang mungkin ada dari proses pembelajaran.

5. LESSON LEARNED

Pada bagian ini Anda menuliskan Lesson Learned (**dokumentasi masalah dan solusi**) yang diperoleh kelompok dari masing-masing anggota kelompok. Dokumentasi lesson learned ini dianjurkan dalam format log book sebagai keterangan tambahan mengenai solusi dari masalah yang Anda hadapi saat proses pengerjaan Tugas Besar untuk diambil hikmah ketika menghadapi masalah yang sama di waktu yang akan datang.

DAFTAR PUSTAKA

Pada bagian ini Anda menuliskan daftar referensi dalam penyusunan laporan Tugas Besar. Gunakan aplikasi pengelola daftar pustaka, misal Mendeley.