



BAB II

TINJAUAN PUSTAKA

BAB II

TINJAUAN PUSTAKA

II.1. Penelitian Terkait

Adapun penelitian terkait yang akan digunakan sebagai sumber acuan yang relevan dan terkini yaitu:

Berdasarkan penelitian yang dilakukan oleh Ekkal Prasetyo, dkk; (2021) yang berjudul “Implementasi Waterfall Model Dalam Pengembangan Sistem Informasi Laporan penjualan Penduduk”, Kebutuhan informasi menjadi sangat penting terutama dalam proses pengambilan keputusan. Salah satu cara yang dapat digunakan yaitu menggunakan teknologi informasi dan komunikasi seperti sistem informasi. sistem informasi sendiri memiliki berbagai macam diantaranya yaitu sistem informasi laporan penjualan yang banyak digunakan untuk menyajikan informasi kepada pihak laporan penjualan dalam mengambil keputusan atau kebijakan. Seperti halnya yang terjadi diberbagai kecamatan saat ini kebutuhan informasi penduduk menjadi sangat penting mengingat dibutuhkan dalam pengambilan kebijakan bagi pemerintah. untuk itu di dalam penelitian ini dibuat sebuah sistem informasi laporan penjualan penduduk guna membantu memetakan keadaan penduduk berdasarkan data kependudukan itu sendiri. untuk menghasilkan sistem informasi laporan penjualan sesuai dengan kebutuhan laporan penjualan maka proses pengembangan digunakan waterfall model dengan lima tahapan pengembangan communication, planning, modeling, construction, dan deployment. Hasil dari pengembangan berupa sistem informasi laporan

penjualan yang dapat memberikan informasi secara visual kepada pihak terkait. Sistem informasi juga berjalan dengan baik yang dibuktikan dari hasil pengujian yang dilakukan.

Berdasarkan penelitian yang dilakukan oleh Dimas A, dkk; (2020) yang berjudul “ Sistem Informasi Laporan penjualan Akademik Menggunakan Data Warehouse Dan Ad-Hoc Query”. Informasi mengenai akademik yang bertambah banyak membuat kampus x memerlukan sebuah Sistem Informasi Laporan penjualan untuk memantau perkembangan akademik dan memiliki fleksibilitas dalam penggunaannya. Hal ini dapat dicapai dengan mengembangkan sistem informasi laporan penjualan ini dengan data warehouse dan ad-hoc query. Berdasarkan hal tersebut, dibuatlah sebuah prototype sistem informasi laporan penjualan yang menggunakan Data Warehouse dan Ad- Hoc Query dalam pembangunan Sistem Informasi Laporan penjualan Akademik pada Kampus X. Adapun metode yang digunakan dalam pengembangan ini adalah The Spiral Model. Spiral Model tepat digunakan dalam pembuatan Sistem Informasi Laporan penjualan Akademik, dikarenakan Sistem Informasi Laporan penjualan Akademik memiliki cakupan yang luas dan Sistem Informasi Laporan penjualan Akademik ini akan berkembang lebih besar lagi untuk memenuhi kebutuhan data dari Laporan penjualan Kampus X. Metode Spiral memiliki tahapan yang harus dilakukan seperti communication, planning, modeling, construction dan deployment. Dalam tahapan tersebut akan memiliki beberapa iterasi yang bergantung pada feedback yang diberikan oleh pengguna sistem tersebut.

Berdasarkan penelitian yang dilakukan oleh Ella W, dkk; (2019) yang

berjudul “Prototype Sistem Informasi Laporan penjualan Dosen di Perguruan Tinggi Swasta ABCD”. Universitas ABCD merupakan salah satu lembaga institusi pendidikan tinggi swasta di Indonesia. Pengelolaan dan peningkatan kualitas di berbagai program studi yang ada menjadi perhatian utama level laporan penjualan, khususnya tenaga pengajar pendidikan tinggi (dosen). Untuk meningkatkan kualitas tenaga pengajar pendidikan tinggi, maka diperlukan sebuah sistem informasi yang membantu laporan penjualan untuk dapat mengakses data dosen, sehingga laporan penjualan dapat mengambil kesimpulan dari data dosen tersebut. Hasil output dari sistem informasi laporan penjualan tersebut berupa chart atau diagram laporan mengenai data dosen. Metode pengembangan sistem yang digunakan ialah model prototype, tahapan metode prototype diantaranya adalah mendengarkan pelanggan, merancang dan membuat prototype kemudian dilakukan uji coba.

II.2. Tinjauan Pustaka

II.2.1. Sistem Informasi

Sistem Informasi merupakan salah satu kebutuhan yang sangat besar akan kemajuan Rumah Sakit. Sistem Informasi adalah kumpulan perangkat keras, perangkat lunak, database, telekomunikasi, manusia dan prosedur yang dirancang untuk men-transformasikan data kedalam bentuk informasi yang berguna. (Mara D, 2017:30)

Sistem merupakan kumpulan elemen – elemen yang saling terkait dan bekerja sama untuk memproses masukan (input) yang ditujukan kepada sistem

tersebut dan mengolah masukan tersebut sampai menghasilkan keluaran (output) yang diinginkan. Suatu sistem pada dasarnya sekelompok unsur – unsur yang erat hubungannya satu dengan yang lain yang berfungsi untuk mencapai tujuan tertentu. Sistem informasi memiliki makna sistem yang bertujuan menampilkan informasi. Ketersediaan komputer telah menjadikan peran informasi berbeda dari zaman sebelumnya. Komputer dapat di manfaatkan untuk mengelola informasi dalam jumlah yang sangat besar dan dalam waktu yang tepat. Saat informasi dikelola secara manual, jumlah dan peran informasi tidak seluar biasa seperti sekarang. Informasi dikelola dalam bentuk sistem, sehingga disebut sistem informasi. (Agustina S, 2019: 13)

Suatu sistem mempunyai karakteristik atau sifat-sifat tertentu, yaitu mempunyai komponen system (component) batas system (boundary) lingkungan luar sistem (environment) penghubung (interface) masukan (input) pengolahan (process) keluaran (output) dan sasaran (objective) atau tujuan (goals). Berikut adalah penjelasan dari karekteristik sistem :

a. Komponen Sistem

Komponen sistem (Companent) yaitu dari sejumlah komponen yang saling berinteraksi yang artinya saling bekerja sama membentuk satu kesatuan. Elemen-elemen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem. Setiap subsistem mempunya sistem untuk menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan, setiap sistem tidak peduli betapa kecilnya pun yang mengandung sub-subsistem.

b. Batas Sistem

Batas Sistem (Boundary) yaitu daerah yang membatasi suatu sistem dengan sistem lainnya atau dengan lingkungan luarnya. Batas sistem ini memungkinkan suatu sistem dipandang.

c. Lingkungan Luar Sistem

Lingkungan luar sistem (Environment) dari suatu sistem yaitu apapun di luar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan dapat juga merugikan sistem tersebut.

d. Penghubung

Penghubung sistem (Interface) merupakan media penghubung antara satu sub system dengan subsistem lainnya, sumber daya-sumber daya mengalir dari subsistem ke subsistem yang lainnya. Dengan penghubung satu subsistem dapat berintegrasi dengan subsistem yang lainnya membentuk satu kesatuan. Masukan Masukan (Input) merupakan energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masukan perawatan (maintenance input) dan masukan sinyal (signal input).

e. Keluaran

Keluaran (Output) Merupakan hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan. Keluaran dapat merupakan masukan untuk subsistem yang lain atau kepada supra sistem.

e. Pengolahan

Pengolahan (Process) yaitu suatu sistem dapat mempunyai suatu bagian pengolah yang akan mengubah masukan menjadi keluaran. Sasaran

(Objectives) dan Tujuan (Goal)

Suatu sistem dikatakan berhasil apabila mengenai sasaran atau tujuannya. Jika suatu sistem tidak mempunyai sasaran, maka operasi sistem tidak akan ada gunanya. Penulis menyimpulkan bahwa sistem adalah kumpulan bagian-bagian atau subsistem-subsistem yang disatukan dan didesain untuk mencapai suatu tujuan. (Rianto S, 2019: 64)

II.2.2. Papan bunga

Usaha papan bunga merupakan kegiatan usaha berbentuk jasa pembuatan rangkaian bunga berbentuk tulisan untuk disewakan kepada konsumen, serta para pelaku usaha papan bunga sering membuat suatu forum yang beranggota para pelaku usaha papan bunga di suatu wilayah tertentu. Salah satu jenis forum pelaku usaha papan bunga di medan adalah Asosiasi Florist Medan yang menaungi GSR Florist. (Ilham B, 2020, 184)

II.2.3. PHP

PHP adalah salah satu server side yang dirancang khusus untuk aplikasi web. PHP disisipkan diantara bahasa HTML dan karena bahasa server side, maka bahasa PHP akan dieksekusi di server, sehingga yang dikirimkan ke browser adalah hasil jadi dalam bentuk HTML, dan kode PHP tidak akan terlihat. PHP termasuk Open Source Product. Jadi, dapat diubah source code dan mendistribusikanya secara bebas. (Priyo S; 2016 : 25).

II.2.4. MySQL

MySQL adalah salah satu jenis database server yang sangat terkenal. Kepopulerannya disebabkan MySQL menggunakan SQL sebagai bahasa dasar

untuk mengakses databasenya. MySQL termasuk jenis RDBMS (Relational Database Management System). Pada MySQL, sebuah database mengandung satu atau sejumlah tabel. Tabel terdiri atas sejumlah baris dan setiap baris mengandung satu atau beberapa kolom. (Priyo S; 2016 : 25).

II.2.5. Database

Basisdata (Database) merupakan kumpulan dari data yang saling berhubungan antara satu dengan yang lain. Basis data atau database merupakan salah satu komponen yang penting dalam sistem informasi, karena berfungsi berfungsi sebagai basis penyedia informasi bagi pemakainya, Sistem basis data adalah suatu sistem informasi yang mengintegrasikan kumpulan dari data yang saling berhubungan dengan yang lainnya dan untuk membuatnya tersedia beberapa aplikasi yang bermacam-macam dalam suatu sistem organisasi. Sistem basis data adalah suatu sistem menyusun dan mengelola record-record menggunakan komputer untuk menyimpan atau merekam serta memelihara data operasional lengkap sebuah organisasi atau perusahaan sehingga mampu menyediakan informasi yang optimal yang diperlukan pemakai untuk proses mengambil keputusan. (Priyo S; 2016 : 25).

II.2.6. Web

Web merupakan terobosan baru sebagai teknologi sistem informasi yang menghubungkan data dari banyak sumber dan layanan yang beragam macamnya di internet, Web cepat sekali populer di lingkungan pengguna internet, karena kemudahan yang diberikan kepada pengguna internet untuk melakukan penelusuran, penjelajahan, dan pencarian informasi.(Priyo S; 2016 : 25).

II.2.7. UML (Unified Modelling Language)

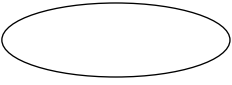
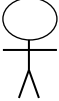
Hasil pemodelan pada OOAD terdokumentasikan dalam bentuk Unified Modeling Language (UML). UML adalah satu alat bantu yang sangat handal di dunia pengembangan sistem yang berorientasi objek. Hal ini disebabkan karena UML menyediakan bahasa pemodelan visual yang memungkinkan bagi pengembang sistem untuk membuat cetak biru atas visi mereka dalam membentuk yang baku, mudah dimengerti serta dilengkapi dengan mekanisme yang efektif untuk berbagi (sharing) dan mengkomunikasikan rancangan mereka dengan yang lain (Adi F ; 2019 : 39).

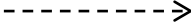
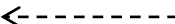
Alat bantu yang digunakan dalam perancangan berorientasi objek berbasis UML adalah sebagai berikut :

1. Use case Diagram

Use case diagram merupakan pemodelan untuk kelakuan (behavior) sistem informasi yang akan dibuat. Use case mendeskripsikan tipikal interaksi antara (pengguna) sebuah sistem dengan sistemnya sendiri melalui sebuah cerita bagaimana sebuah sistem yang dipakai (Munawar ; 2018 : 89).

Tabel II.2. Simbol Use Case Diagram

Gambar	Keterangan
	<i>Use case</i> menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal nama <i>use case</i> .
	Aktor adalah <i>abstraction</i> dari orang atau sistem yang lain yang mengaktifkan fungsi dari target sistem. Untuk mengidentifikasi aktor, harus ditentukan pembagian tenaga kerja dan tugas-tugas yang berkaitan dengan peran pada konteks target sistem. Orang atau sistem bisa muncul dalam beberapa peran. Perlu dicatat bahwa aktor berinteraksi dengan

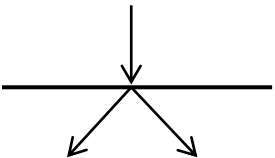
	<i>use case</i> , tetapi tidak memiliki control terhadap <i>use case</i> .
	Asosiasi antara aktor dan <i>use case</i> , digambarkan dengan garis tanpa panah yang mengindikasikan siapa atau apa yang meminta interaksi secara langsung dan bukannya mengidikasikan aliran data.
	Asosiasi antara aktor dan <i>use case</i> yang menggunakan panah terbuka untuk mengidinkasikan bila aktor berinteraksi secara pasif dengan sistem.
	<i>Include</i> , merupakan di dalam <i>use case</i> lain (<i>required</i>) atau pemanggilan <i>use case</i> oleh <i>use case</i> lain, contohnya adalah pemanggilan sebuah fungsi program.
	<i>Extend</i> , merupakan perluasan dari <i>use case</i> lain jika kondisi atau syarat terpenuhi.

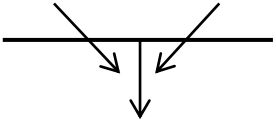
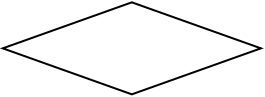
(Sumber : Munawar ; 2018 : 89)

2. Diagram Aktivitas (Activity Diagram)

Activity Diagram bagian penting dari UML yang menggambarkan aspek dinamis dari sistem. Logika prosedural, proses bisnis dan aliran kerja suatu bisnis bisa dengan mudah dideskripsikan dalam activity diagram (Munawar ; 2018 : 137).

Tabel II.3. Simbol Diagram Aktivitas

Gambar	Keterangan
	<i>Start point</i> , diletakkan pada pojok kiri atas dan merupakan awal aktifitas.
	<i>End point</i> , akhir aktifitas.
	<i>Activites</i> , menggambarkan suatu proses/kegiatan bisnis.
	<i>Fork</i> (Percabangan), digunakan untuk menunjukkan kegiatan yang dilakukan secara parallel atau untuk menggabungkan dua kegiatan paralel menjadi satu.

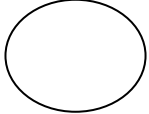
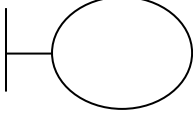
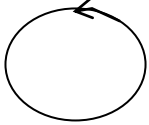
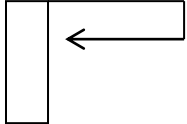
	<i>Join</i> (penggabungan) atau <i>rake</i> , digunakan untuk menunjukkan adanya dekomposisi.
	<i>Decision Points</i> , menggambarkan pilihan untuk pengambilan keputusan, <i>true</i> , <i>false</i> .
	<i>Swimlane</i> , pembagian <i>activity</i> diagram untuk menunjukkan siapa melakukan apa.

(Sumber : Munawar ; 2018 : 137)

3. Diagram Urutan (Sequence Diagram)

Sequence diagram adalah salah satu interaction diagram. Karena sequence diagram mengacu kepada obyek, maka sbelum membuat diagram ini class diagram sudah harus teridentifikasi(Munawar ; 2018 : 186).

Tabel II.4. Simbol Diagram Urutan

Gambar	Keterangan
	<i>EntityClass</i> , merupakan bagian dari sistem yang berisi kumpulan kelas berupa entitas-entitas yang membentuk gambaran awal sistem dan menjadi landasan untuk menyusun basis data.
	<i>Boundary Class</i> , berisi kumpulan kelas yang menjadi <i>interface</i> atau interaksi antara satu atau lebih aktor dengan sistem, seperti tampilan formentry dan <i>form</i> cetak.
	<i>Control class</i> , suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepada entitas, contohnya adalah kalkulasi dan aturan bisnis yang melibatkan berbagai objek.
	<i>Message</i> , simbol mengirim pesan antar <i>class</i> .
	<i>Recursive</i> , menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri

	<i>Activation</i> , <i>activation</i> mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivitas sebuah operasi.
	<i>Lifeline</i> , garis titik-titik yang terhubung dengan objek, sepanjang <i>lifeline</i> terdapat <i>activation</i> .

(Sumber : Munawar ; 2018 : 186)

4. Class Diagram (Diagram Kelas)

Merupakan hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam model desain dari suatu sistem, juga memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem.

Class diagram merupakan diagram statis dari suatu aplikasi. Class Diagram tidak hanya digunakan untuk memvisualisasikan, menggambarkan, dan mendokumentasikan berbagai aspek sistem tetapi juga untuk membangun kode eksekusi (executable code) dari aplikasi perangkat lunak. (Munawar ; 2018 : 101).

Tabel II.5. Simbol Class Diagram

Multiplicity	Penjelasan
1	Satu dan hanya satu
0..*	Boleh tidak ada atau 1 atau lebih
1..*	1 atau lebih
0..1	Boleh tidak ada, maksimal 1
n..n	Batasan antara. Contoh 2..4 mempunyai arti minimal 2 maksimum 4

(Sumber : Munawar ; 2018 : 101)

II.2.8. Normalisasi Data

Normalisasi merupakan sebuah teknik dalam logical desain sebuah basis data yang mengelompokkan atribut dari suatu relasi sehingga membentuk struktur relasi yang baik (tanpa redundansi). Normalisasi adalah proses pembentukan struktur basis data sehingga sebagian besar ambiguity bisa dihilangkan.

Tujuan Normalisasi adalah:

1. Untuk menghilangkan kerangkapan data.
2. Untuk mengurangi kompleksitas.
3. Untuk mempermudah pemodifikasian data.

Jika data dalam database tersebut belum di normalisasi maka akan terjadi 3 kemungkinan yang akan merugikan sistem secara keseluruhan.

INSERT Anomali: Situasi dimana tidak memungkinkan memasukkan beberapa jenis data secara langsung di database.

DELETE Anomali: Penghapusan data yang tidak sesuai dengan yang diharapkan, artinya data yang harusnya tidak terhapus mungkin ikut terhapus.

UPDATE Anomali: Situasi dimana nilai yang diubah menyebabkan inkonsistensi database, dalam artian data yang diubah tidak sesuai dengan yang diperintahkan atau yang diinginkan.

a. Proses Normalisasi

Data diuraikan dalam bentuk tabel, selanjutnya dianalisis berdasarkan persyaratan tertentu ke beberapa tingkat.

Apabila tabel yang diuji belum memenuhi persyaratan tertentu, maka tabel tersebut perlu dipecah menjadi beberapa tabel yang lebih sederhana sampai memenuhi bentuk yang optimal.

Pentingnya Normalisasi

Suatu rancangan database disebut buruk jika:

- a. Data yang sama tersimpan di beberapa tempat (file atau record)
- b. Ketidakmampuan untuk menghasilkan informasi tertentu.

- c. Terjadi kehilangan informasi.
- d. Terjadi adanya redundansi (pengulangan) atau duplikasi data sehingga memboroskan ruang penyimpanan dan menyulitkan saat proses updating data.
- e. Timbul adanya NULL VALUE.
- f. Kehilangan informasi bisa terjadi bila pada waktu merancang database (melakukan proses dekomposisi yang keliru).
- g. Bentuk normalisasi yang sering digunakan adalah 1st NF, 2nd NF, 3rd NF, dan BCNF.

1. Normalisasi Database, 1NF, 2 NF, 3NF

Normalisasi database terdiri dari banyak bentuk, dalam ilmu basis data ada setidaknya 9 bentuk normalisasi yang ada yaitu: 1NF, 2NF, 3NF, EKNF, BCNF, 4NF, 5NF, DKNF, dan 6NF. Namun dalam prakteknya dalam dunia industri bentuk normalisasi yang paling sering digunakan ada sekitar 5 bentuk.

2. Normal Form

Data yang direkam dan dimasukkan secara mentah dalam suatu tabel pada bentuk ini sangat mungkin terjadi inkonsistensi dan anomali data. Guna melakukan normalisasi database kita harus mengidentifikasi data seperti apa yang akan disimpan. Sebagai contoh data dari struk penjualan, seperti terlihat pada tabel dibawah ini.

Kode Faktur	Tanggal	Kode Barang	Nama Barang	Harga Barang	QTY
KD-1001	01/05/2021	BRG-001	Supermie Goreng	2500	8
		BRG-002	Supermie Rebus	2300	9
		BRG-003	Indomie Goreng	2700	10
		BRG-004	Indomie Rebus	2500	5
KD-1002	01/05/2021	BRG-005	Mie Sedap Goreng	2600	7
		BRG-006	Mie Sedap Rebus	2400	8
KD-1003	02/05/2021	BRG-007	Pop Mie	5000	10

Tabel .1 tabel Penjualan

Contoh data pada Tabel 2 merupakan data yang belum dinormalisasi, selanjutnya menuju tahap normalisasi 1NF.

3. Membuat bentuk Normal ke 1 (1NF)

Suatu tabel dikatakan 1NF jika dan hanya jika setiap atribut dari data tersebut hanya memiliki nilai tunggal dalam satu baris. Jadi, tabel yang belum dinormalisasi tadi perlu diubah, sehingga bentuk 1NF, seperti terlihat pada Tabel 2 dibawah ini :

Kode Faktur	Tanggal	Kode Barang	Nama Barang	Harga Barang	QTY
KD-1001	01/05/2021	BRG-001	Supermie Goreng	2500	8
KD-1001	01/05/2021	BRG-002	Supermie Rebus	2300	9
KD-1001	01/05/2021	BRG-003	Indomie Goreng	2700	10
KD-1001	01/05/2021	BRG-004	Indomie Rebus	2500	5
KD-1002	01/05/2021	BRG-005	Mie Sedap Goreng	2600	7
KD-1002	01/05/2021	BRG-006	Mie Sedap Rebus	2400	8
KD-1003	02/05/2021	BRG-007	Pop Mie	5000	10

Tabel 2.2 tabel normal 1NF Penjualan

Inti dari normalisasi 1NF adalah tidak boleh ada grouping data ataupun duplikasi data. Sekarang lanjut pada tahap normalisasi 2NF.

4. Membuat ke Bentuk Normal ke-2 (2NF)

Syarat 2NF adalah tidak diperkenankan adanya partial “functional dependency” Kepada primary key dalam sebuah tabel. Apa itu “functional dependency”? Functional dependency adalah setiap atribut yang bukan kunci (non key) bergantung secara fungsional terhadap primary key. Intinya adalah pada tahap normalisasi 2NF ini tabel tersebut harus dipecah berdasarkan primary key. Sehingga bentuk normalisasi 2NF dari tabel tersebut seperti pada Tabel 3 dan Tabel Hasil Normalisasi Bentuk Normal 2 (2NF).

Tabel 2.3 Tabel Baran

Kode Barang	Nama Barang	Harga
BRG-001	Supermie Goreng	2500
BRG-002	Supermie Rebus	2300
BRG-003	Indomie Goreng	2700
BRG-004	Indomie Rebus	2500
BRG-005	Mie Sedap Goreng	2600
BRG-006	Mie Sedap Rebus	2400
BRG-007	Pop Mie	5000

Kode Faktur	Tanggal	Kode Barang	QTY
KD-1001	01/05/2021	BRG-001	8
KD-1001	01/05/2021	BRG-002	9
KD-1001	01/05/2021	BRG-003	10
KD-1001	01/05/2021	BRG-004	5
KD-1002	01/05/2021	BRG-005	7
KD-1002	01/05/2021	BRG-006	8
KD-1003	02/05/2021	BRG-007	10

Setelah dinormalisasi 2NF, tabelnya terpecah menjadi 2. Sekarang lanjut pada tahap normalisasi 3NF.

5. Bentuk Normal ke 3 (3NF)

Pada 3NF tidak diperkenankan adanya partial “transitive dependency” dalam sebuah tabel. Apa itu “transitive dependency”? Transitive dependency biasanya terjadi pada tabel hasil relasi, atau kondisi dimana terdapat tiga atribut A, B, C. Kondisinya adalah $A \Rightarrow B$ dan $B \Rightarrow C$. Maka C dikatakan sebagai transitive dependency terhadap A melalui B. Intinya pada 3NF ini, jika terdapat suatu atribut yang tidak bergantung pada primary key tapi bergantung pada field yang lain maka atribut-atribut tersebut perlu dipisah ke tabel baru. Contohnya ada pada atribut qty, kolom tersebut tidak bergantung langsung pada primary key kode_faktur melainkan bergantung pada kolom kode_barang. Jadi setelah dinormalisasi 3NF akan menghasilkan Tabel 5, Tabel 6 dan Tabel 7.

Kode Barang	Nama Barang	Harga
BRG-001	Supermie Goreng	2500
BRG-002	Supermie Rebus	2300
BRG-003	Indomie Goreng	2700
BRG-004	Indomie Rebus	2500
BRG-005	Mie Sedap Goreng	2600
BRG-006	Mie Sedap Rebus	2400
BRG-007	Pop Mie	5000

Tabel 2.5. Bentuk Normal ke-3 (3NF) untuk Tabel Barang

Kode Faktur	Tanggal
KD-1001	01/05/2021
KD-1002	01/05/2021
KD-1003	02/05/2021

Kode Faktur	Kode Barang	QTY	Harga
KD-1001	BRG-001	8	2500
KD-1001	BRG-002	9	2300
KD-1001	BRG-003	10	2700
KD-1001	BRG-004	5	2500
KD-1002	BRG-005	7	2600
KD-1002	BRG-006	8	2400
KD-1003	BRG-007	10	5000

Tabel 2.7. Bentuk Normal ke-3 (3NF) untuk Tabel Detail Barang

Hasil Normalisasi 3NF dapat dilihat pada tahap normalisasi 3NF menghasilkan 1

tabel baru dari hasil pemecahan tabel transaksi (Tabel 6). yaitu tabel detail barang (Tabel 7) yang isinya menampung barang-barang yang dibeli. Perhatikan Tabel 7, kenapa di tabel detail barang terdapat kolom harga lagi? padahal kolom harga sudah ada di tabel barang. Kolom harga pada tabel detail barang digunakan untuk menyimpan harga barang pada saat proses transaksi. Jadi, meskipun kolom harga pada tabel barang berubah (naik/turun), harga barang yang ada pada tabel detail barang tidak ikut berubah (fixed). Bayangkan jika kita tidak menambahkan kolom harga pada tabel detail barang, maka yang terjadi total invoice dari transaksi akan berubah seiring berubahnya harga barang. (Setyowati, 2021 : 22).