

BAB II TINJAUAN PUSTAKA

2.1. Aplikasi *Mobile*

Aplikasi *mobile* berasal dari kata *application* dan *mobile*. *Application* yang artinya penerapan, lamaran, penggunaan. Secara istilah aplikasi adalah program siap pakai yang dibuat untuk melaksanakan suatu fungsi bagi pengguna atau aplikasi yang lain dan dapat digunakan oleh sasaran yang dituju sedangkan *mobile* dapat diartikan sebagai perpindahan dari suatu tempat ke tempat yang lain. Secara lebih lengkap, aplikasi *mobile* adalah program siap pakai yang melaksanakan fungsi tertentu yang dipasang pada perangkat *mobile*. [1]

Aplikasi adalah program yang digunakan orang untuk melakukan sesuatu pada sistem komputer. *Mobile* dapat diartikan sebagai perpindahan yang mudah dari satu tempat ke tempat yang lain, misalnya telepon yang dapat berpindah dengan mudah dari satu tempat ke tempat lain tanpa terjadi pemutusan atau terputusnya komunikasi. Sistem aplikasi *mobile* merupakan aplikasi yang dapat digunakan walaupun pengguna berpindah dengan mudah dari satu tempat ke tempat lain tanpa terjadi pemutusan atau terputusnya komunikasi. Aplikasi ini dapat diakses meliputi perangkat nirkabel seperti pager, seperti telepon seluler dan PDA. [2]

Adapun karakteristik perangkat *mobile* yaitu : [2]

1. Ukuran yang kecil : perangkat *mobile* memiliki ukuran yang kecil. Konsumen menginginkan perangkat yang terkecil untuk kenyamanan dan mobilitas mereka.
2. *Memory* yang terbatas : perangkat *mobile* juga memiliki *memory* yang kecil, yaitu *primary* (RAM) dan *secondary* (*disk*).
3. Daya proses yang terbatas: sistem *mobile* tidaklah setangguh *desktop*.
4. Mengonsumsi daya yang rendah : perangkat *mobile* menghabiskan sedikit daya dibandingkan dengan mesin *desktop*.
5. Kuat dan dapat diandalkan : karena perangkat *mobile* selalu dibawa kemana saja, mereka harus cukup kuat untuk menghadapi benturan-benturan, gerakan dan sesekali tetesan-tetesan air.
6. Konektivitas yang terbatas : perangkat *mobile* memiliki *bandwith* rendah, beberapa dari mereka bahkan tidak tersambung.

2.2. Android

Android adalah sistem operasi berbasis Linux yang dimodifikasi untuk perangkat bergerak (*mobile devices*) yang terdiri dari sistem operasi, *middleware*, dan aplikasi-aplikasi utama. Awalnya, *Android* dikembangkan oleh *Android Inc*, perusahaan ini kemudian dibeli oleh Google pada tahun 2005. Sistem operasi Android kemudian diluncurkan bersamaan dengan dibentuknya organisasi Open Handset Alliance pada tahun 2007. Android telah mengalami sejumlah pembaruan sejak pertama kali dirilis. Rata-rata, versi terbaru dari Android dirilis setiap 6 bulan. [3]

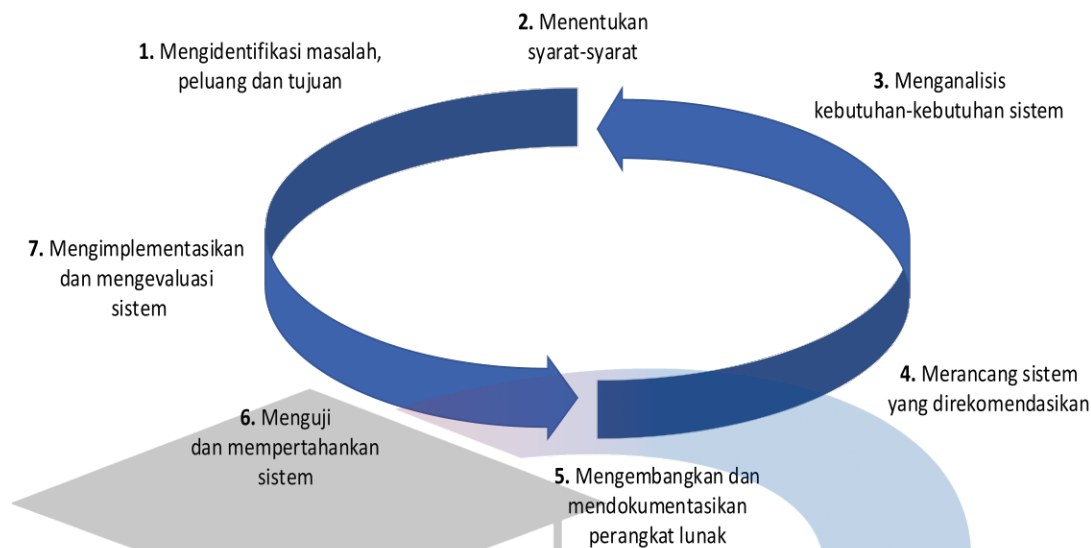
2.3. User Interface

Sebuah aplikasi atau program dikatakan mudah digunakan jika menggunakan tampilan yang baik. Pengguna akan memberikan tanggapan yang positif berupa *feedback* dari aplikasi yang dapat mempermudah proses yang dilakukan. Selain itu user interface merupakan sistem yang kompleks karena dikendalikan oleh pengguna dan merupakan tahap persiapan rancang bangun dari implementasi. Antar muka pengguna sendiri menurut Larasati (2010) selalu dikaitkan dengan tampilan layar, sebab desain yang baik menjadi indikator yang terpenting untuk membuat pengguna merasa tertarik menggunakan website atau aplikasi tersebut. [4] Tujuan dari antar muka pengguna (*user interface*) adalah untuk memungkinkan pengguna menjalankan setiap tugas dalam kebutuhan pengguna (*user requirement*). Jadi, dalam membangun sebuah antar muka pengguna harus berdasar pada kebutuhan pengguna (*user*). [5]

2.4. System Development Life Cycle (SDLC)

Metodologi pengembangan sistem khususnya sistem informasi secara umum menggunakan metodologi *System Development Life Cycle* (SDLC). SDLC adalah pendekatan melalui beberapa tahap untuk menganalisis dan merancang sistem yang dimana sistem tersebut telah dikembangkan dengan sangat baik melalui penggunaan siklus kegiatan penganalisis dan pemakai secara spesifik. [6]

SDLC meliputi tujuh fase yang memiliki masing-masing ciri yang berbeda dan tetap berada dalam kesatuan yang tak terpisahkan. Berikut merupakan gambar dan siklus tahapan-tahapan yang ada dalam SDLC.



Gambar 2. 1 Tahapan SDLC

Berikut merupakan tahapan-tahapan yang ada dalam SDLC : [6]

1. Mengidentifikasi masalah, peluang dan tujuan

Tahap pertama ini sangat penting bagi keberhasilan proyek bahwa penganalisis harus melihat dengan jujur pada apa yang terjadi didalam bisnis. Kemudian bersama dengan anggota organisasional lain, penganalisis akan menentukan dengan tepat masalah-masalah yang ada pada bisnis. Peluang adalah situasi dimana penganalisis yakin bahwa peningkatan bisa dilakukan melalui penggunaan sistem informasi terkomputerisasi. Mengukur peluang memungkinkan bisnis untuk mencapai sisi kompetitif atau menyusun standar-standar industri.

Mengidentifikasi tujuan juga menjadi komponen terpenting dimana penganalisis harus menemukan apa yang sedang dilakukan dalam bisnis. Kemudian penganalisis akan bisa melihat beberapa aspek dalam aplikasi-aplikasi sistem informasi untuk membantu bisnis mencapai tujuannya dengan menyebutkan *problem* atau peluang-peluang tertentu.

2. Menentukan syarat-syarat informasi

Tahapan ini penganalisis memasukkan apa saja yang menentukan syarat-syarat informasi untuk para pemakai yang terlibat. Orang-orang yang terlibat dalam tahap ini adalah penganalisis dan pemakai, biasanya manajer operasi dan pegawai operasional. Penganalisis sistem perlu tahu detail-detail fungsi-fungsi sistem yang ada : siapa (orang-orang yang terlibat), apa (kegiatan bisnis), dimana (lingkungan dimana

pekerjaan itu dilakukan), kapan (waktu yang tepat), dan bagaimana (bagaimana prosedur yang harus dijalankan) dari bisnis yang sedang dipelajari.

3. Menganalisis kebutuhan sistem

Tahap ini penganalisis akan menganalisis kebutuhan-kebutuhan sistem seperti perangkat dan teknik-teknik tertentu yang akan membantu penganalisis dalam menentukan kebutuhan bisnis. Perangkat yang dimaksud ialah penggunaan diagram aliran data untuk menyusun daftar input, proses dan output fungsi bisnis dalam bentuk grafik terstruktur. Penganalisis sistem juga menganalisis keputusan terstruktur yang dibuat. Keputusan terstruktur adalah keputusan-keputusan dimana kondisi, kondisi alternatif, tindakan serta aturan tindakan ditetapkan.

4. Merancang sistem sistem yang direkomendasikan

Dalam tahap ini, penganalisis sistem menggunakan informasi-informasi yang terkumpul sebelumnya untuk mencapai desain sistem informasi yang logik. Bagian dari perancangan sistem informasi yang logik adalah peralatan antarmuka pengguna. Antarmuka menghubungkan pemakai dengan sistem. Contoh dari antarmuka pemakai adalah keyboard (untuk mengetik pertanyaan dan jawaban), menu-menu pada layar (untuk mendatangkan perintah pemakai), serta berbagai jenis *Graphical user Interfaces* (GUIs) yang menggunakan mouse atau cukup dengan sentuhan pada layar.

Tahap perancangan juga mencakup perancangan file-file atau basis data yang bisa untuk menyimpan data-data yang diperlukan seperti basis data yang tersusun dengan baik dan merancang *layout*, *input*, *output* spesifikasi *file*, dan detail-detail proses, serta pohon keputusan atau tabel, aliran data, *flowchart* sistem serta nama-nama dan fungsi-fungsi sub program.

5. Mengembangkan dan mendokumentasikan perangkat lunak

Pada tahap kelima ini penganalisis dapat menggunakan berbagai macam perangkat untuk merancang dan mendokumentasikan perangkat lunak, diantaranya adalah rencana struktur, *Nassi-Shneiderman charts*, dan *pseudocode*.

6. Menguji dan mempertahankan sistem

Pada tahap ini sebelum sistem informasi dapat digunakan, maka harus dilakukan pengujian terlebih dahulu. Sebagian pengujian dilakukan oleh pemrogram sendiri, dan lainnya dilakukan oleh penganalisis sistem. Mempertahankan sistem dan

dokumentasinya dimulai ditahap ini dan dilakukan secara rutin selama sistem informasi dijalankan.

7. Mengimplementasikan dan mengevaluasi sistem

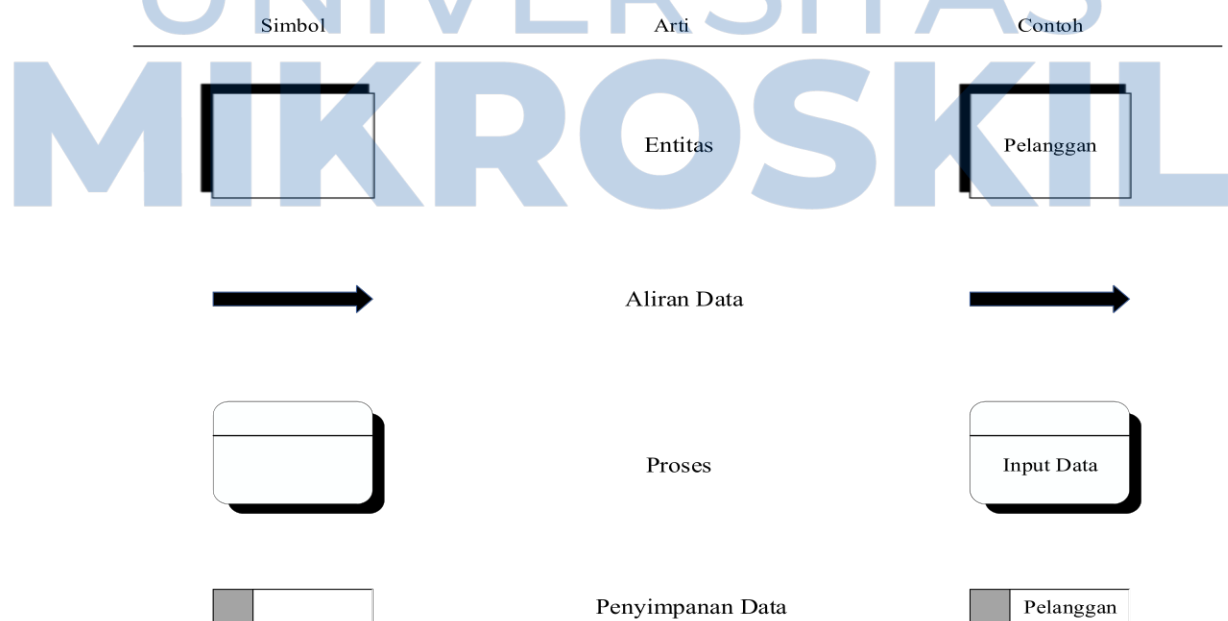
Pada tahap terakhir dari pengembangan sistem, penganalisis membantu untuk mengimplementasikan sistem informasi. Tahap ini melibatkan pelatihan bagi pemakai untuk mengendalikan sistem. Evaluasi yang ditujukan sebagai bagian dari tahapan rancangan akhir siklus hidup pengembangan sistem.

Kerja sistem biasanya berulang, ketika penganalisis menyelesaikan satu tahap pengembangan sistem maka akan berlanjut ke tahap berikutnya. Penemuan suatu masalah bisa memaksa penganalisis kembali ke tahap sebelumnya dan memodifikasi pekerjaannya di tahap tersebut.

2.5. Alat Bantu Pengembangan Sistem

2.5.1. Data Flow Diagram (DFD)

Data flow diagram (DFD) adalah teknik analisis terstruktur, penganalisis sistem dapat mempresentasikan proses-proses data didalam organisasi. Pendekatan DFD menekankan logika yang mendasari sistem dengan menggunakan kombinasi dari empat simbol, penganalisis sistem dapat membuat penggambaran proses yang bisa menampilkan dokumentasi sistem yang solid. [6]



Gambar 2. 2 Simbol-simbol pada DFD

Beberapa kelebihan dari DFD adalah : [6]

1. Kebebasan dari menjalankan implementasi teknis sistem.
2. Pemahaman lebih jauh mengenai keterkaitan satu sama lain dalam sistem dan subsistem.
3. Mengkomunikasikan pengetahuan sistem yang ada dengan pengguna melalui diagram aliran data.
4. Menganalisis sistem yang diajukan untuk menentukan apakah data-data dan proses yang diperlukan sudah ditetapkan.

2.5.2. Kamus Data

Kamus data adalah suatu aplikasi khusus dari jenis kamus-kamus yang digunakan sebagai referensi kehidupan setiap hari. Kamus data merupakan hasil referensi data mengenai data, data yang disusun oleh penganalisis sistem untuk membimbing mereka selama melakukan analisis dan desain. [6]

Kamus data otomatis berguna dikarenakan memiliki kapasitas dalam hal referensi silang item-item data, dengan demikian kamus data sangat penting untuk sistem-sistem besar karena mampu menghasilkan ribuan elemen data yang dikatalogkan dan dibuat referensi silang. [6]

Adapun notasi aljabar yang digunakan dalam pembuatan kamus data, yaitu: [6]

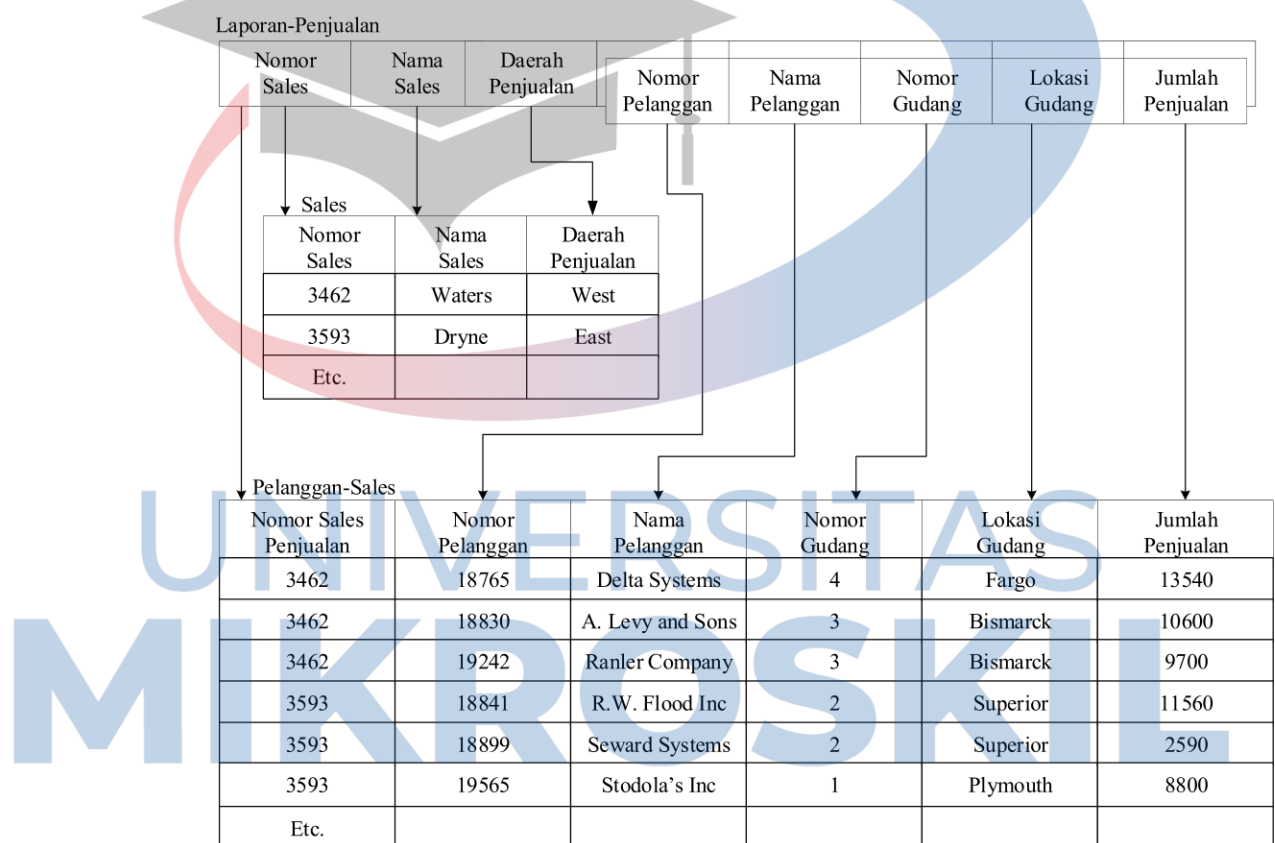
1. Tanda sama dengan (=), artinya “terdiri dari”.
2. Tanda plus (+), artinya “dan”.
3. Tanda kurung {}, artinya menunjukkan elemen-elemen repetitif atau kelompok berulang. Kelompok berulang bisa mengandung keadaan-keadaan tertentu seperti jumlah perulangan yang pasti atau batas tertinggi dan batas terendah untuk jumlah pengulangan.
4. Tanda kurung [], artinya menunjukkan salah satu dari dua situasi tertentu. Satu elemen bisa ada sedangkan elemen lainnya juga ada, tetapi tidak bisa keduanya ada secara bersamaan. Elemen-elemen yang ada didalam tanda kurung ini saling terpisah satu sama lain.
5. Tanda kurung (), artinya suatu elemen yang bersifat pilihan. Elemen-elemen yang bersifat pilihan ini bisa dikosongkan pada layar masukan atau bisa juga memuat spasi atau nol untuk *field-field* numerik pada struktur file.

2.5.3. Normalisasi

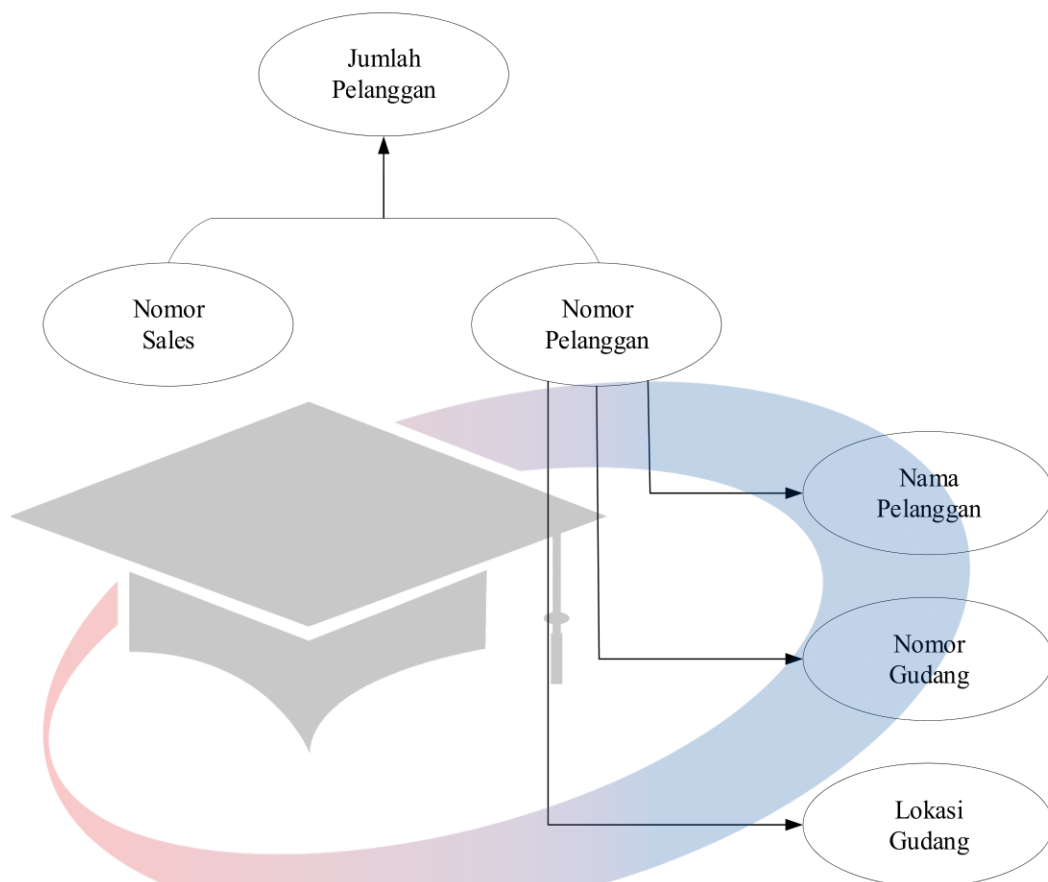
Normalisasi adalah transformasi tinjauan pemakai yang kompleks dan data tersimpan ke sekumpulan bagian-bagian struktur data yang kecil dan stabil. Di samping menjadi lebih sederhana dan lebih stabil, struktur data yang dinormalisasikan lebih mudah diatur daripada struktur data lainnya. [6]

Pada proses normalisasi, terdapat bentuk-bentuk normalisasi yaitu : [6]

1. Bentuk normalisasi pertama (1NF) langkah pertama dalam normalisasi hubungan adalah menghilangkan kelompok terulang dan mengidentifikasi kunci utama. Untuk mengerjakan perlu dipecah kedalam dua atau lebih hubungan.



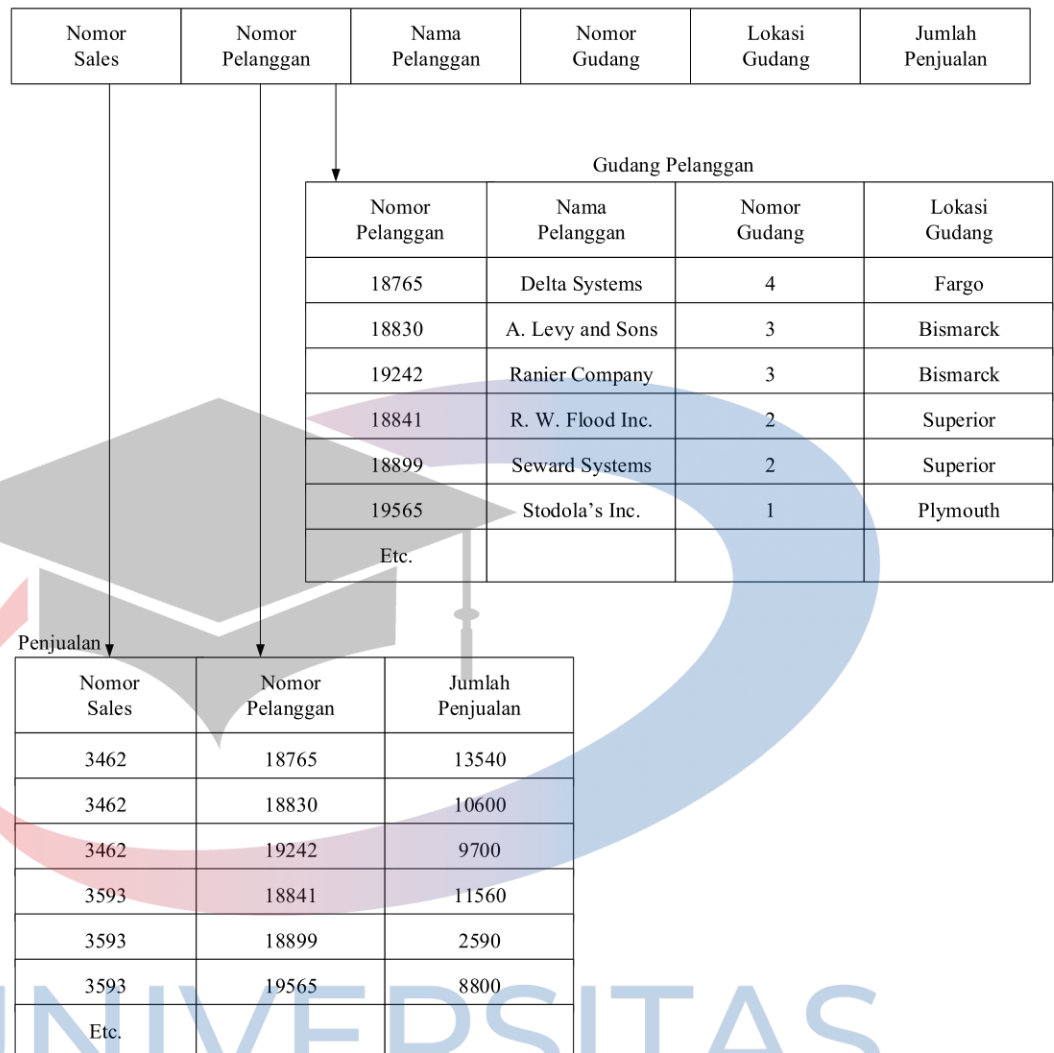
Gambar 2. 3 Bentuk Normalisasi 1NF



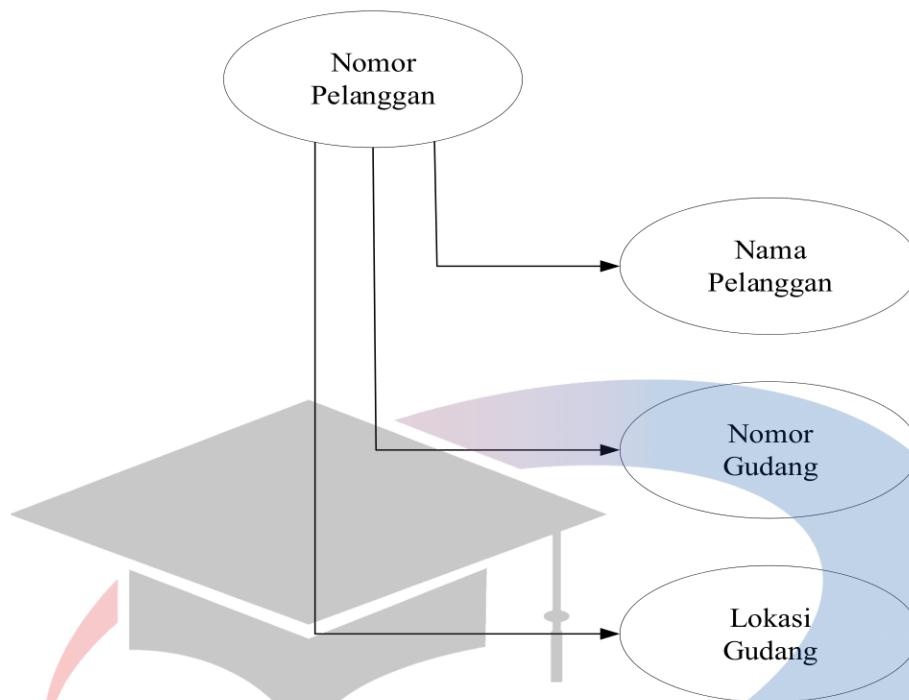
Gambar 2. 4 Diagram model data

2. Bentuk normalisasi kedua (2NF) pada tahap ini menjamin bahwa semua atribut bukan kunci sepenuhnya tergantung pada kunci utama.

UNIVERSITAS
MIKROSKIL

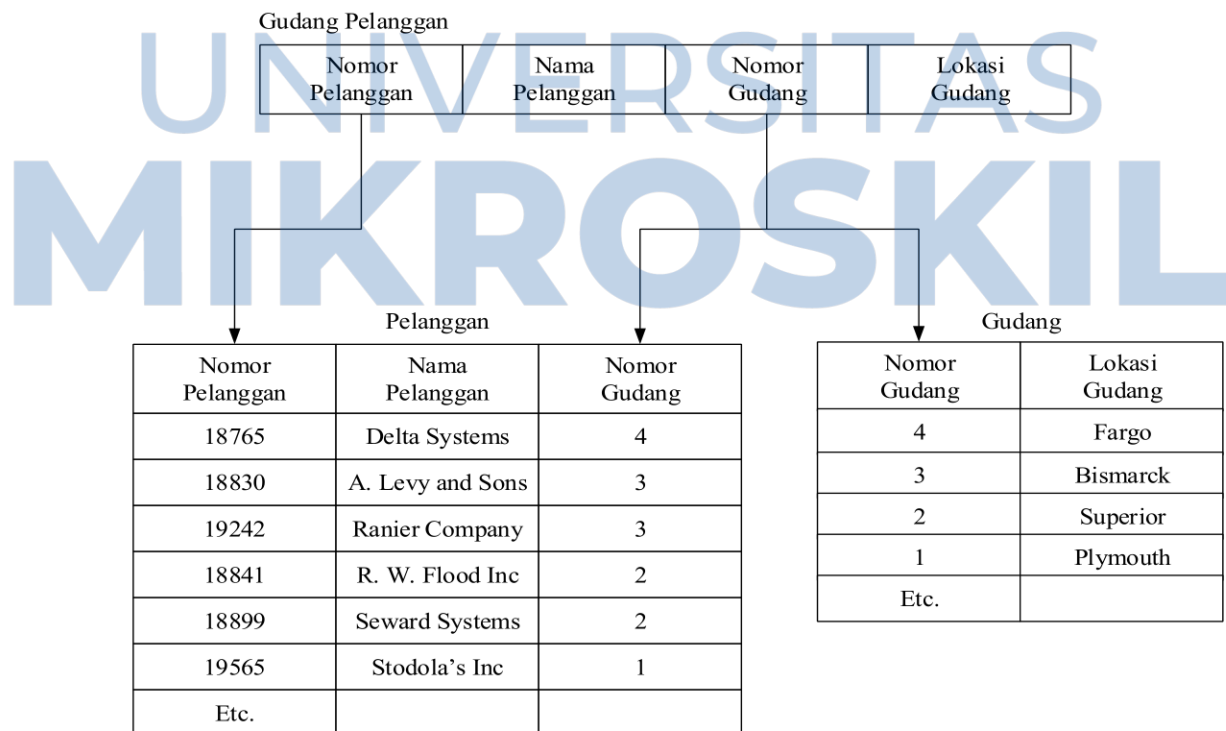


Gambar 2. 5 Bentuk Normalisasi 2NF



Gambar 2. 6 Diagram model data ketergantungan transitif

3. Bentuk normalisasi ketiga (3NF) ditahap ketiga mengubah ketergantungan transitif. Suatu ketergantungan transitif adalah sesuatu dimana atribut bukan kunci tergantung pada atribut bukan kunci lainnya.



Gambar 2. 7 Bentuk Normalisasi 3NF

2.5.4. Basis Data

Basis data tidak hanya merupakan kumpulan file. Lebih dari itu, basis data adalah pusat sumber data yang caranya dipakai oleh banyak pemakai untuk berbagai aplikasi. Inti dari basis data adalah *database management system* (DBMS), yang membolehkan pembuatan, modifikasi dan pembaharuan basis data; mendapatkan kembali data; dan membangkitkan laporan. Orang yang memastikan bahwa basis data memenuhi tujuannya disebut administrator basis data. [6]

Tujuan basis data yang efektif termuat dibawah ini : [6]

1. Memastikan bahwa data dapat dipakai di antara pemakai untuk berbagai aplikasi.
2. Memelihara data baik keakuratan maupun kekonsistennannya.
3. Memastikan bahwa semua data yang diperlukan untuk aplikasi sekarang dan yang akan datang akan disediakan dengan cepat.
4. Membolehkan basis data untuk berkembang dan kebutuhan pemakai untuk berkembang.
5. Membolehkan pemakai untuk membangun pandangan personalnya tentang data tanpa memperhatikan cara data disimpan secara fisik.

2.6. Pemesanan

Pemesanan atau *Taking Order* di restoran menurut Sumarno (2009) adalah kegiatan menerima dan mencatat pesanan tamu. Dalam hal ini makanan dan minuman, yang selanjutnya akan diteruskan ke bagian yang terkait, antara lain dapur, bar, dan kasir. *Taking Order* meliputi beberapa kegiatan antara lain : [7]

1. Menampilkan informasi yang akurat mengenai semua makanan dan minuman yang tersedia di daftar menu.
2. Mencatat menu yang dipesan, jumlah yang dipesan, nama pemesan, dan lainnya.
3. Mengkonfirmasi pesanan kepada pelanggan.
4. Meneruskan pesanan ke bagian terkait.

Pemesanan dalam bahasa *Inggris* lainnya menurut Suartana (1987) adalah *reservation* yang berasal dari kata “*to reserve*” yaitu menyediakan atau mempersiapkan tempat sebelumnya. Pengertian reservasi adalah sebuah proses perjanjian berupa pemesanan sebuah produk baik barang maupun jasa dimana pada saat itu telah terdapat kesepakatan antara konsumen dengan produsen mengenai

produk tersebut, namun belum ditutup oleh sebuah transaksi jual beli. Pada saat reservasi berlangsung biasanya ditandai dengan adanya proses tukar-menukar informasi antara konsumen dan produsen agar kesepahaman mengenai produk dapat terwujud. [8]

2.7. Jasa *Catering*

Catering berasal dari kata *cater* menurut Wulandari (2017) yang berarti menyiapkan dan menyajikan makanan. Biasanya *catering* dibutuhkan dalam sebuah acara. [9] *Catering* bermula dari sebutan *to cater* yang berisi terjemahan bebasnya berarti buat persiapan dan menyajikan makanan dan minuman itu disebut *caterer*. Industri jasa makanan (*katering industri*) meliputi daerah-daerah, institusi beserta perusahaan yang memberikan makanan. Industri ini meliputi restoran, sekolah beserta kafetaria tempat tinggal sakit. [10]

Catering yakni segmen multifaset penyajian makanan industri tersedia banyak bentuk usaha *katering* dalam segmen *katering*. Industri jasa makanan digolongkan menjadi tiga klasifikasi umum yaitu segmen komersial, segmen nonkomersial, beserta segmen militer. [10]

UNIVERSITAS
MIKROSKIL