

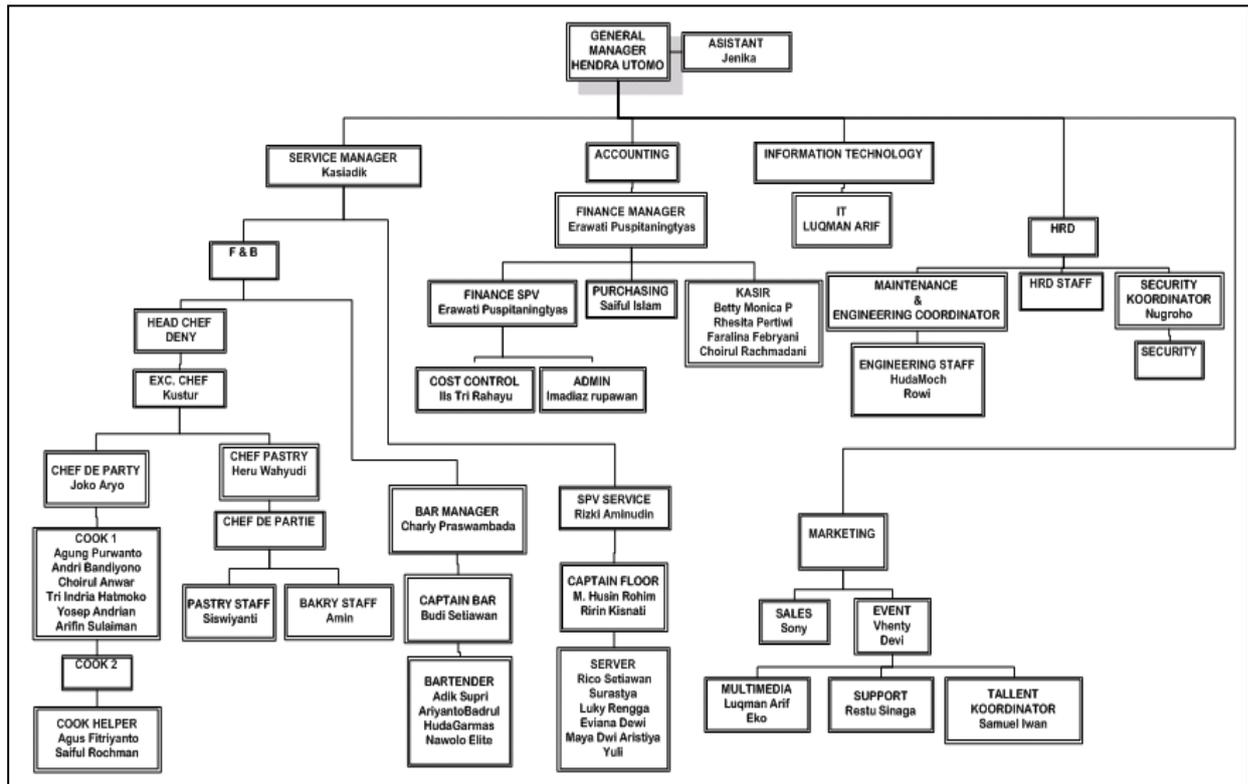
BAB II

TINJAUAN PUSTAKA

Sebagai penunjang, di bawah ini akan disajikan sejumlah dasar-dasar teori yang digunakan sebagai dasar dalam penyusunan Tugas Akhir ini.

2.1 Sekilas Tentang Perusahaan

Resto De'kasteel adalah salah satu usaha yang bergerak dibidang *food & baverage*, Restoran dan cafe yang bertempat di jalan Villa Taman Telaga yang terletak di daerah Citraland, Proses pelayanan terhadap *customer* restoran ini melalui beberapa system manajemen yang telah di terapkan oleh Resto De'kasteel itu sendiri, besarnya jumlah penjualan produk menu masakan diharapkan seimbang sehingga bisa memenuhi omzet yang di targetkan. Pemenuhan kualitas pelayanan terhadap *cusomer* adalah sesuatu yang menjadi fokus utama dari Resto De'kasteel yang berdiri tepat 16 September tahun 2010. Pelayanan yang di berikan haruslah bisa mencapai kepuasan terhadap *customer* restoran dikarenakan kepuasan dan kenyamanan pelanggan adalah modal utama dari Resto De'kasteel. Tindakan yang dilakukan oleh pihak manajemen Resto De'kasteel dalam rangka pemenuhan kualitas pelayanan adalah dengan cara memeriksa kebersihan, gizi pada setiap menu yang di sajikan dan kepuasan *customer* akan pelayanan yang di berikan. Pemeriksaan kualitas masakan di Resto De'kasteel di lakukan untuk menjaga kualitas kebersihan serta gizi yang seimbang pada menu masakan sehingga kualitas menu masakan tidak kalah dibandingkan dengan restoran manapun.



Gambar 2.1 Struktur Organisasi Resto De'kasteel

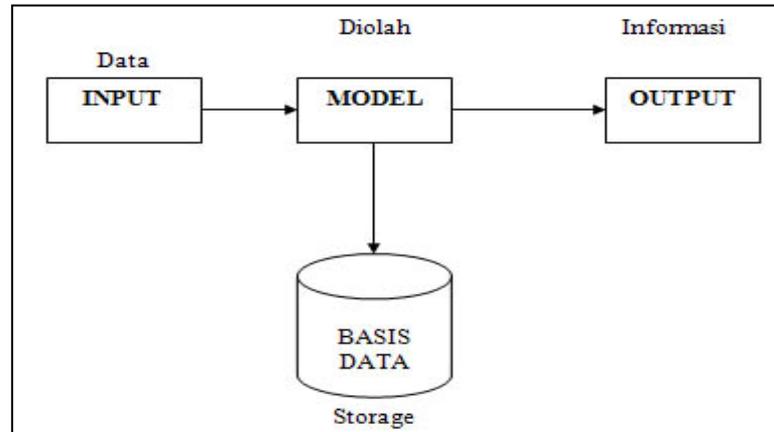
2.2 Sistem Informasi

Sistem informasi merupakan suatu sistem yang tujuannya memberikan informasi. Informasi adalah data yang diolah menjadi bentuk yang berguna bagi para pemakaiannya. Agar informasi tersebut berguna, maka informasi harus didukung oleh tiga pilar yaitu: tepat kepada orangnya atau relevan (*relevance*), tepat waktu (*timeliness*) dan tepat nilainya atau akurat (*accurate*). *Output* yang tidak didukung oleh ketiga pilar ini tidak dapat dikatakan sebagai informasi yang berguna, tetapi merupakan sampah (*garbage*).

Tugas sistem informasi adalah untuk melakukan siklus pengolahan data dimana untuk melakukan siklus ini, maka sebagai suatu sistem diperlukan

komponen-komponen tertentu. Siklus ini disebut dengan siklus pengolahan data (*data processing life cycle*) atau disebut juga dengan nama siklus informasi (*information life cycle*).

Untuk melakukan siklus pengolahan data diperlukan tiga buah komponen, yaitu komponen *input*, komponen model dan komponen *output*. Data yang masih belum diolah perlu disimpan untuk pengolahan lebih lanjut, karena tidak semua data yang diperoleh disimpan terlebih dahulu yang nantinya setiap saat dapat diambil untuk diolah menjadi informasi. Data ini disimpan di simpanan (*storage*) dalam bentuk basis data (*database*). Proses pengolahan data dari basis data yang diolah menjadi informasi disebut dengan *extended data processing life cycle* yang ditunjukkan pada Gambar 2.2 di bawah ini:



Gambar 2.2 *Extended Data Processing Life Cycle*

Basis Data (*database*) adalah kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan pada perangkat keras (*hardware*) komputer

dan dapat digunakan pada perangkat lunak (*software*) untuk memanipulasi data-datanya. Dari konsep dasar sistem informasi dan basis data dapat dilakukan manajemen/pengolahan data informasi sehingga data dapat diproses sewaktu-waktu ketika dibutuhkan karena sistem informasi menggunakan sistem *storage*/penyimpanan data dalam sebuah *database*.

2.2.1 Konsep Dasar Sistem Informasi

Suatu sistem pada dasarnya adalah sekelompok unsur yang erat hubungannya satu dengan yang lain, yang berfungsi bersama-sama untuk mencapai tujuan tertentu. Secara sederhana, suatu sistem dapat diartikan sebagai suatu kumpulan atau himpunan dari unsur, komponen, atau variabel yang terorganisir, saling berinteraksi, saling tergantung satu sama lain dan terpadu. Dari definisi ini dapat dirinci lebih lanjut pengertian sistem secara umum, yaitu:

- a. Setiap sistem terdiri dari unsur-unsur.
- b. Unsur-unsur tersebut merupakan bagian terpadu sistem yang bersangkutan.
- c. Unsur sistem tersebut bekerja sama untuk mencapai tujuan sistem.
- d. Suatu sistem merupakan bagian dari sistem lain yang lebih besar.

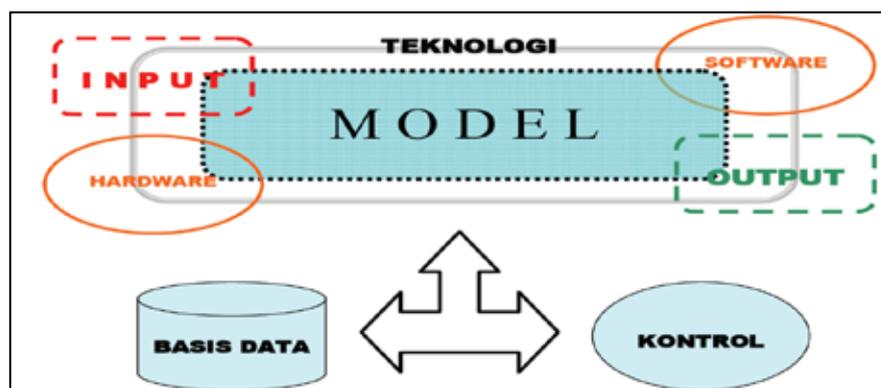
Secara umum informasi dapat didefinisikan sebagai hasil dari pengolahan data dalam suatu bentuk yang lebih berguna dan lebih berarti bagi penerimanya yang menggambarkan suatu kejadian-kejadian yang nyata yang digunakan untuk pengambilan keputusan. Informasi merupakan data yang telah diklasifikasikan atau diolah atau diinterpretasi untuk digunakan dalam proses pengambilan keputusan.

Sistem informasi adalah suatu sistem dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan informasi yang diperlukan untuk pengambilan keputusan.

Sistem informasi dalam suatu organisasi dapat dikatakan sebagai suatu sistem yang menyediakan informasi bagi semua tingkatan dalam organisasi tersebut kapan saja diperlukan. Sistem ini menyimpan, mengambil, mengubah, mengolah dan mengkomunikasikan informasi yang diterima dengan menggunakan sistem informasi atau peralatan sistem lainnya.

2.2.2 Komponen Sistem Informasi

Sistem informasi terdiri dari komponen-komponen yang disebut blok bangunan (*building block*), yang terdiri dari komponen *input*, komponen model, komponen *output*, komponen teknologi, komponen *hardware*, komponen *software*, komponen basis data, dan komponen kontrol. Semua komponen tersebut saling berinteraksi satu dengan yang lain membentuk suatu kesatuan untuk mencapai sasaran.



Gambar 2.3 Interaksi Sistem Informasi

Penjelasan untuk komponen sistem informasi yang ada pada gambar di atas adalah sebagai berikut:

a. Komponen *input*

Input mewakili data yang masuk ke dalam sistem informasi. *Input* disini termasuk metode dan media untuk menangkap data yang akan dimasukkan, yang dapat berupa dokumen-dokumen dasar.

b. Komponen model

Komponen ini terdiri dari kombinasi prosedur, logika, dan model matematik yang akan memanipulasi data *input* dan data yang tersimpan di basis data dengan cara yang sudah ditentukan untuk menghasilkan keluaran yang diinginkan.

c. Komponen *output*

Hasil dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua pemakai sistem.

d. Komponen teknologi

Teknologi merupakan "*tool box*" dalam sistem informasi, Teknologi digunakan untuk menerima *input*, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran dan membantu pengendalian dari sistem secara keseluruhan.

e. Komponen *hardware*

Hardware berperan penting sebagai suatu media penyimpanan vital bagi sistem informasi. Yang berfungsi sebagai tempat untuk menampung *database* atau lebih mudah dikatakan sebagai sumber data dan informasi untuk memperlancar dan mempermudah kerja dari sistem informasi.

f. Komponen *software*

Software berfungsi sebagai tempat untuk mengolah, menghitung dan memanipulasi data yang diambil dari *hardware* untuk menciptakan suatu informasi.

g. Komponen basis data

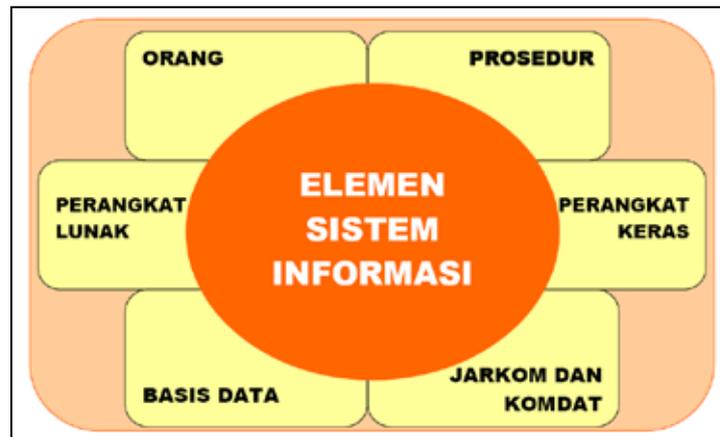
Basis data (*database*) merupakan kumpulan data yang saling berkaitan dan berhubungan satu dengan yang lain, tersimpan di perangkat keras komputer dan menggunakan perangkat lunak untuk memanipulasinya. Data perlu disimpan dalam basis data untuk keperluan penyediaan informasi lebih lanjut. Data di dalam basis data perlu diorganisasikan sedemikian rupa supaya informasi yang dihasilkan berkualitas. Organisasi basis data yang baik juga berguna untuk efisiensi kapasitas penyimpanannya. Basis data diakses atau dimanipulasi menggunakan perangkat lunak paket yang disebut DBMS (*Database Management System*).

h. Komponen kontrol

Banyak hal yang dapat merusak sistem informasi, seperti bencana alam, api, air, debu, kecurangan-kecurangan, kegagalan-kegagalan sistem itu sendiri, ketidak efisienan, sabotase dan lain sebagainya. Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dapat dicegah ataupun bila terlanjur terjadi sebuah kesalahan dapat langsung cepat diatasi.

2.2.3 Elemen Sistem Informasi

Sistem informasi terdiri dari elemen-elemen yang terdiri dari orang, prosedur, perangkat keras, perangkat lunak, basis data, jaringan komputer dan komunikasi data. Semua elemen ini merupakan komponen fisik.



Gambar 2.4 Hubungan Elemen Sistem Informasi

Penjelasan untuk masing-masing elemen sistem informasi adalah sebagai berikut:

a. Orang

Orang atau personil yang dimaksud yaitu operator komputer, analis sistem, *programmer*, personil data entry, dan manajer sistem informasi/EDP.

b. Prosedur

Prosedur merupakan elemen fisik. Hal ini disebabkan karena prosedur disediakan dalam bentuk fisik seperti buku panduan dan instruksi. Ada 3 jenis prosedur yang dibutuhkan, yaitu instruksi untuk pemakai, instruksi untuk penyiapan masukan, instruksi pengoperasian untuk karyawan pusat komputer.

c. Perangkat keras

Perangkat keras bagi suatu sistem informasi terdiri atas komputer (pusat pengolah, unit masukan/keluaran), peralatan penyiapan data, dan terminal masukan/keluaran.

d. Perangkat lunak

Perangkat lunak dapat dibagi dalam 3 jenis utama:

- a) Sistem perangkat lunak umum, seperti sistem pengoperasian dan sistem manajemen data yang memungkinkan pengoperasian sistem komputer.
- b) Aplikasi perangkat lunak umum, seperti model analisis dan keputusan.
- c) Aplikasi perangkat lunak yang terdiri atas program yang secara spesifik dibuat untuk setiap aplikasi.

e. Basis data

File yang berisi program dan data dibuktikan dengan adanya media penyimpanan secara fisik seperti *disket*, *harddisk*, *magnetic tape*, dan sebagainya. File juga meliputi keluaran tercetak dan catatan lain diatas kertas, mikro film, dan lain sebagainya.

f. Jaringan komputer

Jaringan komputer adalah sebuah kumpulan komputer, printer dan peralatan lainnya yang terhubung dalam satu kesatuan. Informasi dan data bergerak melalui kabel-kabel atau tanpa kabel sehingga memungkinkan pengguna jaringan komputer dapat saling bertukar dokumen dan data.

g. Komunikasi data

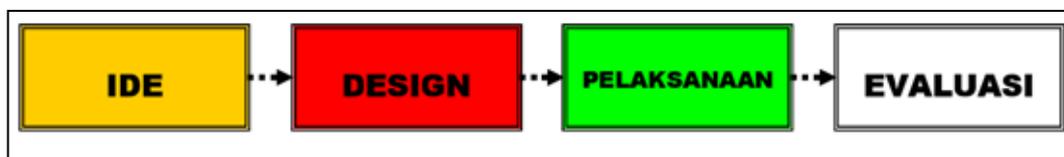
Komunikasi data adalah merupakan bagian dari telekomunikasi yang secara khusus berkenaan dengan transmisi atau pemindahan data dan informasi

diantara komputer-komputer dan piranti-piranti yang lain dalam bentuk digital yang dikirimkan melalui media komunikasi data. Data berarti informasi yang disajikan oleh isyarat digital. Komunikasi data merupakan bagian vital dari suatu sistem informasi karena sistem ini menyediakan infrastruktur yang memungkinkan sebuah komputer dapat berkomunikasi satu sama lain.

2.2.4 Arsitektur Sistem Informasi

Sistem informasi dapat di bentuk sesuai kebutuhan organisasi masing-masing. Oleh karena itu, untuk dapat menerapkan sistem yang efektif dan efisien diperlukan perencanaan, pelaksanaan, pengaturan, dan evaluasi sesuai keinginan masing-masing organisasi. Guna dari sistem yang efektif dan efisien tidak lain untuk mendapatkan keunggulan dalam berkompetisi. Semua orang dapat menggunakan sistem informasi dalam organisasi, tetapi faktor efisiensi setiap sistem adalah berbeda.

Perlu diketahui, perubahan sistem, baik besar maupun kecil, selalu akan melalui tingkatan-tingkatan. Adapun tingkatan yang menjadi kunci yang digunakan untuk memecahkan bagian masalah baik itu secara menyeluruh maupun per bagian, yaitu sebagai berikut :



Gambar 2.5 Tingkatan Perubahan Sistem

Penjelasan dari gambar di atas adalah sebagai berikut:

Tingkat I : Ide, mengetahui perlu adanya perubahan.

Tingkat II : *Design*, merancang cara pemecahannya.

Tingkat III : Pelaksanaan, menerapkan design ke dalam sistem.

Tingkat IV : Kontrol, memeriksa tingkat pelaksanaan dijalankan sesuai dengan design.

Tingkat V : Evaluasi, memeriksa apakah perubahan yang terjadi sesuai dengan tujuan semula.

Tingkat VI : Tindak lanjut, melaksanakan perubahan sesuai dengan hasil evaluasi yang ada.

2.2.5 Klasifikasi Sistem Informasi

Sistem informasi merupakan suatu bentuk integrasi antara satu komponen dengan komponen lain karena sistem memiliki sasaran yang berbeda untuk setiap kasus yang terjadi yang ada di dalam sistem tersebut. Oleh karena itu, sistem dapat di klasifikasikan dari beberapa sudut pandang, diantaranya :

a. Sistem abstrak atau sistem fisik

Sistem abstrak adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik, misalnya sistem teologia, yaitu sistem yang berupa pemikiran hubungan antara manusia dengan Tuhan.

Sistem fisik merupakan sistem secara fisik, misalnya sistem komputer.

b. Sistem alamiah dan sistem buatan manusia

Sistem alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat oleh manusia, misalnya sistem perputaran bumi.

Sistem buatan manusia merupakan sistem yang melibatkan interaksi manusia dengan mesin, yang disebut *human machine system*. Sistem informasi berbasis *internet* merupakan contoh *human machine system* karena menyangkut penggunaan komputer yang berinteraksi dengan manusia.

c. Sistem deterministik dan sistem probabilistik

Sistem deterministik adalah sistem yang beroperasi dengan tingkah laku yang dapat diprediksi.

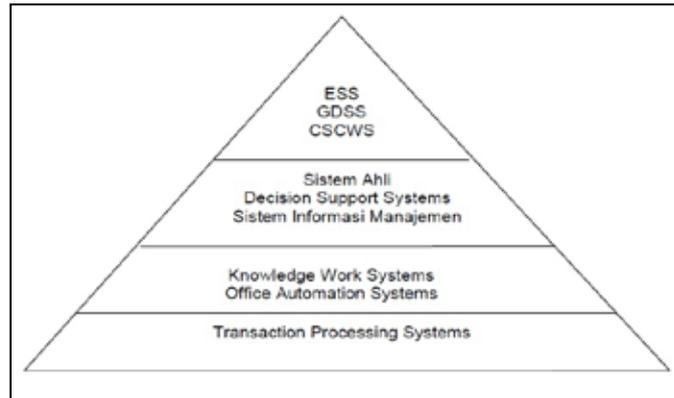
Sistem probabilistik adalah sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur probabilistik.

d. Sistem terbuka dan sistem tertutup

Sistem terbuka adalah sistem yang berhubungan dan dipengaruhi oleh lingkungan luarnya. Sistem ini menerima masukan dan menghasilkan keluaran untuk subsistem lainnya. Sistem tertutup adalah sistem yang tidak terhubung dan tidak terpengaruh oleh lingkungan luarnya. Sistem ini bekerja secara otomatis tanpa campur tangan pihak luar [2].

2.2.6 Jenis-Jenis Sistem Informasi

Sistem informasi dikembangkan untuk tujuan yang berbeda-beda, tergantung pada kebutuhan bisnis. Sistem informasi dapat dibagi menjadi beberapa bagian, yang ditunjukkan pada Gambar 2.6 di bawah ini:



Gambar 2.6 Jenis-jenis Sistem

a. *Transaction Processing Systems (TPS)*

TPS adalah sistem informasi yang terkomputerisasi yang dikembangkan untuk memproses data dalam jumlah besar untuk transaksi bisnis rutin seperti daftar gaji dan inventarisasi. TPS berfungsi pada level organisasi yang memungkinkan organisasi bisa berinteraksi dengan lingkungan eksternal. Data yang dihasilkan oleh TPS dapat dilihat atau digunakan oleh manajer.

b. *Office Automation Systems (OAS) dan Knowledge Work Systems (KWS)*

OAS dan KWS bekerja pada level *knowledge*. OAS mendukung pekerja data, yang biasanya tidak menciptakan pengetahuan baru melainkan hanya menganalisis informasi sedemikian rupa untuk mentransformasikan data atau memanipulasikannya dengan cara-cara tertentu sebelum menyebarkannya secara keseluruhan dengan organisasi dan kadang-kadang diluar organisasi. Aspek-aspek OAS seperti *word processing*, *spreadsheets*, *electronic scheduling* dan komunikasi melalui *voice mail*, *email* dan *video conferencing*. KWS mendukung para pekerja profesional seperti ilmuwan, insinyur dan

doktor dengan membantu menciptakan pengetahuan baru dan memungkinkan mereka mengkontribusikannya ke organisasi atau masyarakat.

c. Sistem Informasi Manajemen (SIM)

SIM tidak menggantikan TPS, tetapi mendukung spektrum tugas-tugas organisasional yang lebih luas dari TPS, termasuk analisis keputusan dan pembuat keputusan. SIM menghasilkan informasi yang digunakan untuk membuat keputusan, dan juga dapat membantu menyatukan beberapa fungsi informasi bisnis yang sudah terkomputerisasi (basis data).

d. *Decision Support Systems (DSS)*

DSS hampir sama dengan SIM karena menggunakan basis data sebagai sumber data. DSS bermula dari SIM karena menekankan pada fungsi mendukung pembuat keputusan diseluruh tahap-tahapnya meskipun keputusan aktual tetap wewenang eksklusif pembuat keputusan.

e. Sistem Ahli (ES) dan Kecerdasan Buatan (AI)

AI dimaksudkan untuk mengembangkan mesin-mesin yang berfungsi secara cerdas. Dua cara untuk melakukan riset AI adalah memahami bahasa alamiahnya dan menganalisis kemampuannya untuk berfikir melalui problem sampai kesimpulan logikanya. Sistem ahli menggunakan pendekatan-pendekatan pemikiran AI untuk menyelesaikan masalah serta memberikannya lewat pengguna bisnis. Sistem ahli (*knowledge-based systems*) secara efektif menangkap dan menggunakan pengetahuan seorang ahli untuk menyelesaikan masalah yang dialami dalam suatu organisasi. Berbeda dengan DSS, DSS meninggalkan keputusan terakhir bagi pembuat keputusan sedangkan sistem ahli menyeleksi solusi terbaik terhadap suatu masalah khusus. Komponen

dasar sistem ahli adalah *knowledge-base* yakni suatu mesin inferensi yang menghubungkan pengguna dengan sistem melalui pengolahan pertanyaan lewat bahasa terstruktur dan antarmuka pengguna.

f. *Group Decision Support Systems (GDSS)* dan *Computer-Support Collaborative Work Systems (CSCW)*

Bila kelompok, perlu bekerja bersama-sama untuk membuat keputusan semi terstruktur dan tak terstruktur, maka *Group Decision Support Systems* membuat suatu solusi. GDSS dimaksudkan untuk membawa kelompok bersama-sama menyelesaikan masalah dengan memberi bantuan dalam bentuk pendapat, kuesioner, konsultasi dan skenario. Kadang-kadang GDSS disebut dengan CSCW yang mencakup pendukung perangkat lunak yang disebut dengan “*groupware*” untuk kolaborasi tim melalui komputer yang terhubung dengan jaringan.

g. *Executive Support Systems (ESS)*

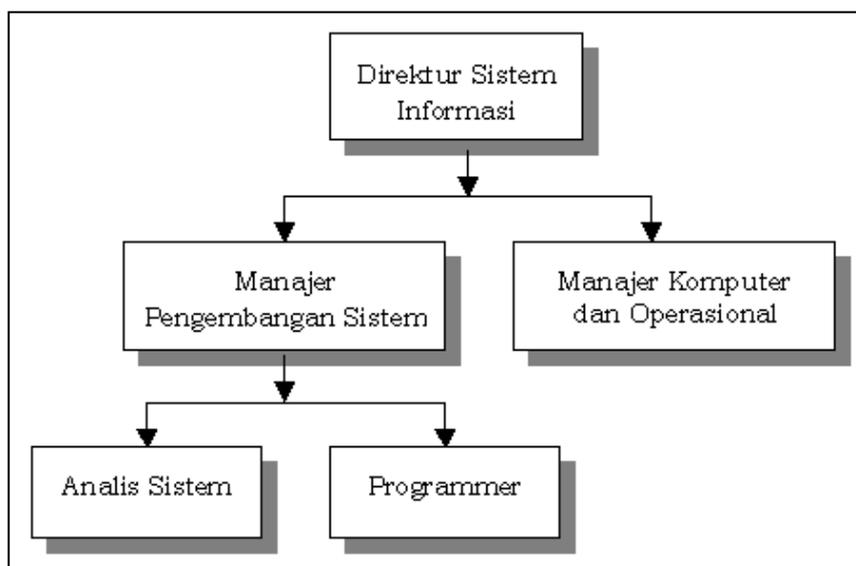
ESS tergantung pada informasi yang dihasilkan TPS dan SIM dan ESS membantu eksekutif mengatur interaksinya dengan lingkungan eksternal dengan menyediakan grafik-grafik dan pendukung komunikasi di tempat-tempat yang bisa diakses seperti kantor [3].

2.2.7 Pengelola Sistem Informasi

Salah satu perangkat yang paling penting dari sistem informasi adalah manusia sebagai pengelola informasi. Oleh karena itu hubungan antara sistem informasi dengan pengelolanya sangat erat. Sistem informasi yang dibutuhkan sangat tergantung dari kebutuhan pengelolanya. Pengelola sistem informasi

terorganisasi dalam suatu struktur manajemen. Oleh karena itu bentuk atau jenis sistem informasi yang diperlukan sesuai dengan level manajemennya [3].

- a. Manajemen Level Atas: untuk perencanaan strategis, kebijakan dan pengambilan keputusan.
- b. Manajemen Level Menengah: untuk perencanaan taktis dan pengambilan keputusan.
- c. Manajemen Level Bawah: untuk perencanaan dan pengawasan operasi dan pengambilan keputusan.
- d. Operator: untuk pemrosesan transaksi dan merespon permintaan.



Gambar 2.7 Contoh Struktur Organisasi Sistem Informasi

Kemudian pada perkembangannya, dengan semakin besarnya lingkup sebuah sistem informasi memerlukan adanya penataan kembali personel dengan baik terutama pada struktur manajemen organisasi personil. Lihat contoh penataan

struktur organisasi pada Departemen Sistem Informasi pada Gambar 2.7. Tetapi struktur organisasi seperti pada gambar tersebut dapat dimodifikasi sesuai dengan kondisi real perusahaan. Variasi struktur manajemen tersebut sangat tergantung pada *Managerial Efficiency* yang dibandingkan dengan tingkat *User Service*.

Manajemen sumber daya manusia perlu dilakukan dengan benar agar sistem informasi dapat berjalan dengan baik. Hal itu dilakukan untuk mengurangi terjadinya kesalahankesalahan yang bersifat manusiawi yang dapat mengurangi mutu informasi yang dihasilkan sebuah sistem. Gordon B. Davis memberikan contoh kesalahan-kesalahan tersebut seperti misalnya:

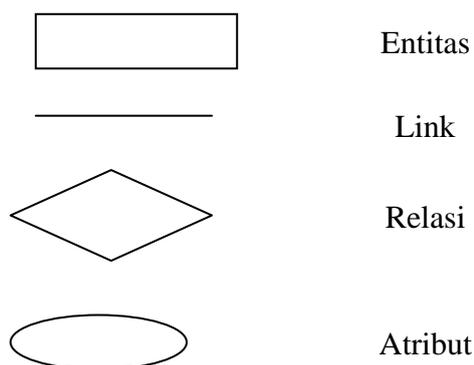
- a. Metode pengumpulan dan pengukuran data yang tidak tepat.
- b. Operator sistem tidak mengikuti prosedur pengolahan yang benar.
- c. Kehilangan data atau data tidak terolah.
- d. Pemeriksaan atau pencatatan data yang salah.
- e. Salah dalam menggunakan dokumen induk/file induk.
- f. Kesalahan dalam prosedur pengolahan.
- g. Kesalahan yang dilakukan dengan sengaja.

Kesulitan karena kesalahan dapat diatasi dengan dua teknik yaitu pengontrolan data dan penambahan batas kepercayaan pada data. Pengontrolan secara intern dapat dilakukan untuk menemukan kesalahan-kesalahan yang terjadi. Selain itu perlu juga dilakukan pemeriksaan auditing baik secara intern maupun ekstern.

2.3 ER Diagram (Entity Relation Diagram)

Yang dimaksud dengan *entity relation diagram* adalah penggambaran relasi antar entitas secara keseluruhan dengan menggunakan grafik atau gambar.

Entity Relation Diagram dapat digambarkan dengan simbol-simbol sebagai berikut:



Gambar 2.1 Simbol dalam ER Diagram

Pada model data relationship hubungan antar file direlasikan dengan kunci relasi, yang merupakan kunci utama dari masing-masing file. Relasi antar dua file atau dua tabel dapat dikategorikan menjadi tiga macam. Relasi antar file tersebut dapat dipisahkan sebagai berikut:

1. *One to One Relationship*

Hubungan antara himpunan entitas pertama dengan himpunan entitas kedua adalah satu berbanding satu. Misalnya satu Dosen Mengepalai satu jurusan .



Gambar 2.2 *One to One Relationship*

2. *One to Many Relationship*

Hubungan antar himpunan entitas pertama dan kedua adalah satu berbanding banyak atau dapat pula dibalik banyak lawan satu. Misal satu dosen mengajar banyak matakuliah



Gambar 2.3 *One to Many Relationship*

3. *Many to Many Relationship*

Hubungan antara himpunan entitas pertama dan kedua adalah banyak berbanding banyak. Misal mahasiswa mana saja boleh mengambil banyak matakuliah dan mahasiswa diajar oleh banyak dosen pula.



Gambar 2.4 *Many to Many Relationship*

Pada *tools* PowerDesigner yang digunakan penulis dalam perancangan dan pembuatan sistem, simbol-simbol yang digunakan pada ER diagram konvensional berbeda dengan simbol-simbol yang digunakan oleh *tools*

PowerDesigner. Pada tabel dibawah ini merupakan simbol-simbol ER diagram yang digunakan oleh penulis dalam pembuatan sistem dengan menggunakan *tools* PowerDesigner

Tabel 2.1 *Simbol ER Diagram (PowerDesigner)*

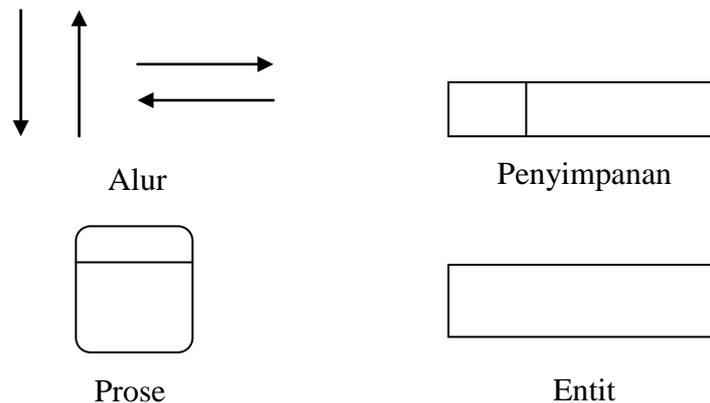
Simbol	Keterangan
	<p>Simbol entitas, “Ent_1” merupakan nama dari entity.</p> <p>Sedangkan “Atribut_1”, “Atribut_2” dan “Atribut_3” merupakan atribut-atribut yang ada pada entity</p>
Simbol	Keterangan
	<p>Simbol <i>one to one relationship</i>, “Relation_11” merupakan nama dari <i>relationship</i></p>
	<p>Simbol <i>one to many relationship</i>, “Macam dana” merupakan nama dari <i>relationship</i></p>
	<p>Simbol <i>many to many relationship</i>, “Melaksanakan” merupakan nama dari <i>relationship</i></p>

2.4 Data Flow Diagram

Data flow diagram adalah gambaran sistem secara logikal. Gambaran ini tidak tergantung pada perangkat keras, perangkat lunak, struktur data atau organisasi file. Pemakai akan mengetahui aliran data dalam suatu sistem dan beserta tempat penyimpanan. Keuntungan menggunakan *data flow diagram*

adalah memudahkan pemakai atau *user* yang awam di bidang komputer untuk mengerti sistem yang akan dibuat.

Simbol - simbol yang digunakan adalah:



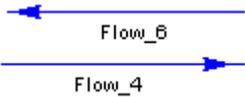
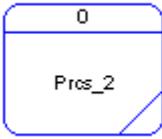
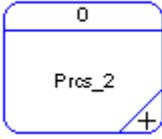
Gambar 2.5 *Simbol Data Flow Diagram*

Data flow diagram ini adalah salah satu alat pembuatan model yang sering digunakan, khususnya bila fungsi-fungsi sistem merupakan bagian yang lebih penting dan kompleks dari pada data yang dimanipulasi oleh sistem. Dengan kata lain, *Data flow diagram* adalah alat pembuatan model yang memberikan penekanan hanya pada fungsi sistem. *Data flow diagram* ini merupakan alat perancangan sistem yang berorientasi pada alur data dengan konsep dekomposisi dapat digunakan untuk penggambaran analisa maupun rancangan sistem yang mudah dikomunikasikan oleh profesional sistem kepada pemakai maupun pembuat program.

Dalam perancangan sistem, penulis menggunakan *tools* Power Designer sebagai media pembantu dalam pembuatan sistem. Pada tabel dibawah ini

merupakan simbol-simbol *Data Flow Diagram* yang digunakan oleh penulis dalam pembuatan sistem dengan menggunakan *tools* PowerDesigner

Tabel 2.2 Simbol Data Flow Diagram (*PowerDesigner*)

Simbol	Keterangan
	Simbol Entity, “Entt_1” merupakan nama dari entity
	Simbol Arus Data, “Flow_4” atau “Flow_6” merupakan nama dari arus data
	Simbol Proses, angka nol “0” menjelaskan tentang identifikasi dari proses, sedangkan “Prccs_2” merupakan nama proses. Proses ini tidak mempunyai sub proses, karena tidak ada tanda “+” pada sudut kiri bawah proses
	Simbol ini merupakan simbol proses seperti yang dijelaskan sebelumnya, bedanya proses ini memiliki sub proses, karena pada sudut kiri bawah terdapat tanda “+”
	Simbol Data Store, angka satu “1” merupakan identifikasi dari data store, sedangkan “Stor_3” merupakan nama dari data store

2.5 PHP

Script PHP (*Hypertext Preprocessor*) merupakan bahasa *web serverside* yang bersifat *open source*. Bahasa PHP menyatu dengan script HTML (*Hyper Text Markup Language*) yang sepenuhnya dijalankan pada *server* [4].

Kelebihan dari bahasa pemrograman PHP adalah sebagai berikut:

- a. Sebuah script yang tidak melakukan kompilasi dalam penggunaannya.
- b. *Web Server* yang mendukung PHP dapat ditemukan dimana-mana dari mulai IIS sampai dengan dengan Apache, dengan konfigurasi yang sangat mudah.
- c. Dalam sisi pengembangan lebih mudah, karena banyaknya *developer* yang membantu dalam pengembangannya.
- d. Pada sisi pemahaman, PHP adalah bahasa scripting yang paling mudah dikuasai karena referensi yang mudah ditemukan.

PHP adalah bahasa pemrograman *Open Source* yang dapat digunakan diberbagai Sistem Operasi (Unix, Linux, Windows) dan dapat dijalankan secara runtime melalui console ataupun terminal serta dapat dijalankan dengan perintah-perintah sistem.

2.6 HTML (*HyperText Markup Language*)

HTML adalah sebuah bahasa *markup* yang digunakan untuk membuat sebuah halaman *web* dan menampilkan berbagai informasi di dalam sebuah *browser internet*. Bermula dari sebuah bahasa yang sebelumnya banyak digunakan di dunia penerbitan dan percetakan yang disebut dengan SGML, HTML adalah sebuah standar yang digunakan secara luas untuk menampilkan

halaman *web* dan HTML kini merupakan standar *internet* yang saat ini dikendalikan oleh *World Wide Web Consortium (W3C)*.

Versi terakhir dari HTML adalah HTML 4.01, meskipun saat ini telah berkembang XHTML yang merupakan pengembangan dari HTML. HTML berupa kode-kode tag yang menginstruksikan *browser* untuk menghasilkan tampilan sesuai dengan yang diinginkan. Sebuah file yang merupakan file HTML dapat dibuka dengan menggunakan *web browser* seperti *Mozilla Firefox* atau *Microsoft Internet Explorer*. HTML juga dapat dikenali oleh aplikasi pembuka email ataupun dari PDA dan program lain yang memiliki kemampuan *browser*.

Salah satu ciri *web* yang dinamis adalah menyediakan form dapat dimanfaatkan pengunjung untuk mengirim informasi kepada *server*. Dengan demikian, ada interaksi dua arah antara pengunjung dan *webmaster* (pemilik *website*), sehingga suatu *website* menjadi lebih hidup.

Dalam pembuatan form ini. Yang perlu diperhatikan adalah tag <FORM> beserta atributnya, seperti *method*, *action*, dan *input* data yang lainnya. *Action* adalah atribut yang menunjukkan isian atau informasi, bahwa data akan dikirimkan ke suatu file proses data yang akan diakses. Atribut *method* mempunyai dua macam pilihan nilai, yaitu POST dan GET.

- a. Jika menggunakan pilihan GET, maka semua isian dikirimkan ke *server* dalam satu kesatuan URL-nya.
- b. Jika menggunakan pilihan POST, data isian akan dikirimkan ke *server* terpisah dengan URL-nya.

Kelemahan dari *method* GET adalah data yang digunakan sebagai paramater akan tampil pada URL, selain itu *method* GET tidak dapat mengirim

data dalam jumlah yang besar. Kemudian yang tidak kalah pentingnya adalah name (nama) dari inputan data, karena nama dari kontrol data ini akan menjadi variabel yang membawa data isian dari form.

2.7 XHTML

XHTML (*eXtensible HyperText Markup Language*) adalah sebuah kombinasi antara HTML (*HyperText Markup Language*) dan XML (*eXtensible HTML Language*) yang merupakan bentuk sempurna dari HTML. Pada penulisan XHTML memiliki aturan lebih ketat dan mengharuskan penulisan tag bersih dibandingkan dengan HTML.

XHTML merupakan bentuk HTML yang didefinisikan sebagai aplikasi XML yang artinya XHTML berisi semua elemen dengan HTML 4.01 dikombinasikan dengan sintaks XML. XML adalah markup language dimana semua penulisan tag harus dibuat secara benar agar dapat menghasilkan dokumen yang memiliki susunan tag kode baik. XML di desain untuk mendiskripsikan data, dan HTML di desain untuk menampilkan data.

Perkembangan teknologi browser, dimana kini tidak hanya ada di komputer saja (browser juga ada di *PDA, handphone, Game Console*, dsb), memunculkan kebutuhan adanya sebuah script yang standard dan bebas dari kesalahan agar dapat berjalan dengan baik disemua *platform* alat penanganannya.

Kombinasi HTML dan XML (yang lalu dikenal dengan nama XHTML) menggunakan aturan yang ketat dalam hal penulisan. Halaman XHTML dapat dibaca pada peralatan yang mendukung XML. Keuntungan XML adalah dapat menyajikan dokumen “*well-formed*” dan support di banyak *browser*.

2.7.1 Perbedaan Antara XHTML dan HTML

Sebenarnya, XHTML tidak terlalu berbeda dengan HTML 4.01. Jika sudah memiliki kemampuan HTML 4.0.1, maka akan sangat membantu dalam memulai XHTML. Perbedaannya adalah :

- a. Dalam XHTML, semua elemen harus saling tersarang dan harus selalu ditutup. Maksudnya harus saling tersarang adalah semua elemen harus ditulis dengan tag pembuka dan ditutup berurutan dengan penutupnya lebih dahulu. Contohnya : `<p> Paragraph ini ditulis dengan cetak tebal yang dimulai dengan tag dan diakhiri oleh tag . </p>`
Paragraph ini ditulis dengan cetak tebal yang dimulai dengan tag `` dan diakhiri oleh tag ``.
- b. Sekalipun dan tampil di browser hanya dengan menuliskan `<p>` tanpa menutupnya dengan `</p>`, hal tersebut bukan merupakan penulisan XHTML yang benar.
- c. Elemen XHTML harus ditulis dengan huruf kecil (*lowercase*).

Jika dalam HTML nya bebas untuk menuliskan dengan:

```
<HTML><HEAD>
<TITTLE> ini adalah tittle situs saya </TITTLE> </HEAD>
<BODY>
<P ALIGN="CENTER">Ini adalah pargraph align center </P>
</BODY></HTML>
```

Dalam XHTML nya harus mengubah dengan huruf kecil semua, berikut script yang akan digunakan:

```
<html><head>
```

```

<title>ini adalah tittle situs saya</title></head>

<body>

<p align="center"> Ini adalah paragraph align center </p>

</body></html>

```

- d. Dokumen XHTML harus mempunyai satu *root* elemen XHTML harus bersarang diantara elemen *root* <html>

2.7.2 Valid XHTML

Valid XHTML adalah sebuah penanda yang diberikan kepada setiap *website/blog* oleh konsorium W3C yang menjelaskan kepada setiap bahwa *website* yang sedang dikunjungi sudah menggunakan kode-kode programan HTML lanjutan (XHTML) yang sudah valid. Valid disini berarti kode yang ditulis sudah sesuai dengan aturan-aturan yang ada dalam standart penulisan XHTML (HTML lanjut). Beberapa aturan dasar untuk valid XHTML adalah :

- a) Penulisan tag XHTML menggunakan huruf kecil, tidak seperti standart penulisan HTML jaman dulu, sekalipun kode yang dituliskan sama persis.
- b) Setiap kode XHTML yang dibuka harus ditutup kembali. Sebagai contoh : jika melakukan pembukaan dengan tag <body> maka harus kembali menutupnya ketika sudah selesai dengan tag <body> dengan </body>.
- c) Setiap tag yang tidak memiliki penutup harus ditutup dengan dirinya sendiri. Beberapa diantaranya adalah :
 - a. penulisan tag
, dimana yang benar adalah

 - b. Penulisan tag , dimana yang benar adalah
 - c. Penulisan tag <hr>, dimana yang benar adalah <hr/>

- d) Setiap tag `` ataupun `` harus dimulai dengan tag `` dan ditutup dengan tag `` setelah `` ataupun ``.
- e) Tag yang mendefinisikan tag lebih general dituliskan belakangan. Sebagai contoh : tidak boleh menuliskan `<center> <p>` ini adalah paragraph teks yang berposisi center `</p> </center>`. Yang benar harus menuliskannya menjadi:
- ```
<p align="center">Ini adalah paragraph teks yang berposisi center</p>
```

## 2.8 MySQL

Dalam aplikasi Sistem Informasi Pemesanan Online di Resto De'kateel Berbasis *Web* dan *SMS Gateway* akan menggunakan MySQL sebagai *database*.

MySQL adalah *database* yang menghubungkan script PHP, menggunakan perintah query dan escape character yang sama dengan PHP. MySQL mempunyai tampilan client yang mempermudah dalam meng-akses *database* dengan kata sandi untuk mengijinkan proses yang boleh dilakukan. Kelebihan dari MySQL dapat melakukan transaksi dengan mudah dan efisien serta mampu menangani jutaan *user* dalam waktu yang bersamaan.

### 2.9.1 Tipe data MySQL

#### a. Tipe Data *Numeric*

Pada tipe data numerik MySQL menggunakan standart ANSI SQL, maka jika bekerja dengan sistem *database* yang berbeda akan tetap dikenal.

**Tabel 2.1** Tipe Data Numerik

Tipe Data	Nilai
TINYINT()	-128 sampai 127 normal 0 sampai 255 UNSIGNED
SMALLINT()	-32768 sampai 32767 normal 0 sampai 65535 UNSIGNED
INT()	-2147483648 sampai 2147483647 normal 0 sampai 4294967295 UNSIGNED
BIGINT()	-9223372036854775808 sampai 9223372036854775807 normal 0 sampai 18446744073709551615 UNSIGNED
FLOAT(M,D)	Bilangan real presisi – tunggal
DOUBLE(M,D)	Bilangan real presisi – ganda
DECIMAL( , )	Nilai sampai +/-79,228 x 10/24

### b. Tipe Data *String*

Walaupun tipe *numeric* dan *date* sangat penting, namun kebanyakan dari tipe data yang akan digunakan berada di format *String*. Pada Tabel 2.2 akan menampilkan beberapa tipe data *string* di dalam MySQL.

**Tabel 2.2** Tipe Data *String*

Tipe Data	Nilai	Format
DATE	1000-01-01 sampai 9999-12-31	YYYY-MM-DD
TIME	-838:59:59 sampai 838:59:59	Hh:mm:ss
DATETIME	1000-01-01 00:00:00 sampai 9999-12-31 23:59:59	YYYY-MM-DD hh:mm:ss
TIMESTAMP	19700101000000 sampai 2037+	YYYYMMDDhhmmss
YEAR	1900 – 2155	YYYY

### c. Tipe Data *Date and Time*

MySQL memiliki beberapa tipe data yang tersedia untuk menampilkan tanggal dan waktu, diantaranya dapat dilihat pada Tabel 2.3

**Tabel 2.3** Tipe Data *Date and Time*

<b>Tipe Data</b>	<b>Nilai</b>	<b>Status</b>
CHAR(M)	M=0-255 karakter	Binary
VARCHAR (M)	M=0-255 karakter (membaca nilai karakter yang digunakan)	Binary
TINYBLOB	0-255 bytes	Case sensitive
BLOB	0-65,535 bytes	Case sensitive
TINYTEXT	0-255 bytes	Not case sensitive
TEXT	0-65,535 bytes	Not case sensitive
ENUM("value1","value2",.....)	Column is exactly 1 of values 1-255 values	Not case sensitive

### 2.6.2 *Structured Query Language (SQL)*

SQL adalah bahasa standard yang digunakan oleh hampir semua *database* seperti MySQL, Oracle atau Microsoft SQL Server. *Database* menggunakan perintah SQL untuk memodifikasi data di dalam tabel, karena SQL dirancang dengan mudah untuk menguraikan hubungan antara tabel dengan baris. Perintah SQL yang sering digunakan antara lain Insert, Select, Update dan Delete.

#### A. Perintah INSERT

Insert merupakan perintah untuk menambahkan data baru ke dalam tabel

*Sintaks :*

**INSERT INTO** nama\_tabel (fields) **VALUES**(nilai\_fields);

## B. Perintah SELECT

Select merupakan perintah untuk menampilkan data dari tabel. Dalam menampilkan data, dapat pula menambahkan kondisi-kondisi tertentu sesuai dengan kebutuhan.

*Sintaks :*

**SELECT** ekspresi\_dan\_fields **FROM** nama\_tabel

[ **WHERE** kondisi ]

[ **ORDER BY** field [ ASC | DESC ] ]

[ **LIKE** field [ kondisi ] ]

[ **LIMIT** offset, rows ] ;

## C. Perintah UPDATE

Update adalah perintah SQL yang digunakan untuk memodifikasi satu atau lebih field di dalam satu tabel.

*Sintaks :*

**UPDATE** nama\_tabel

**SET** field1='nilai\_baru1',

Field2='nilai\_baru2'

[ **WHERE** kondisi ]

## D. Perintah DELETE

Pada saat menggunakan perintah Delete, harus diperhatikan spesifikasi field yang akan dihapus, karena pada saat menjalankan perintah Delete tidak terdapat pesan konfirmasi sebelumnya yang akan mengakibatkan data terhapus semua.

*Sintaks :*

**DELETE FROM** nama\_tabel

[ **WHERE** kondisi ]

## 2.7 SMS Gateway

SMS Gateway adalah sebuah perangkat lunak yang menggunakan bantuan komputer dan memanfaatkan teknologi seluler yang diintegrasikan guna mendistribusikan pesan-pesan yang di-generate lewat sistem informasi melalui media SMS yang di-handle oleh jaringan seluler. Secara khusus, sistem ini akan memiliki fungsi-fungsi sebagai berikut:

- a. *Message Management dan Delivery*
  - a) Pengaturan pesan yang meliputi manajemen prioritas pesan, manajemen, pengiriman pesan, dan manajemen antrian.
  - b) Pesan yang dilalukan harus sedapat mungkin *fail safe*. Artinya, jika terdapat gangguan pada jaringan telekomunikasi, maka sistem secara otomatis akan mengirim ulang pesan tersebut.

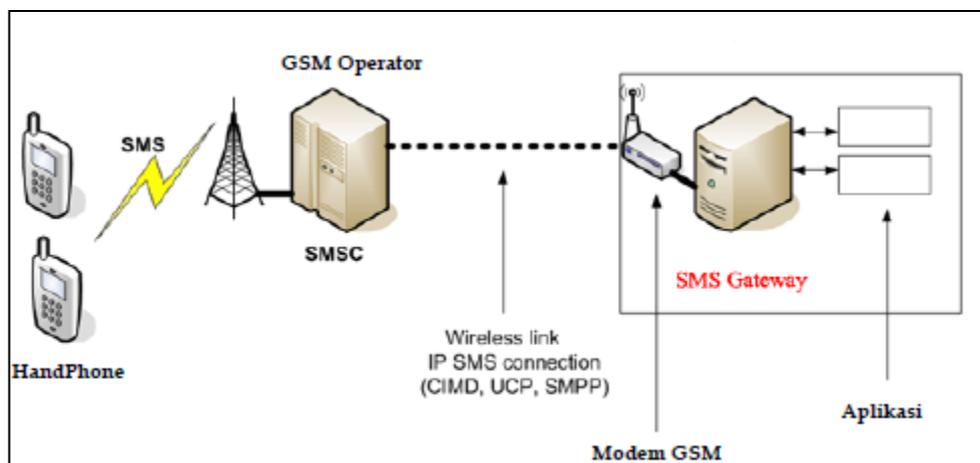
- b. Korelasi

Berfungsi untuk melakukan korelasi data untuk menghasilkan data baru hasil korelasi. Pada sistem yang terpasang saat ini, arsitektur lalu lintas data melalui SMS sudah terjalin cukup baik. Hanya saja, keterbatasan akses data dan tujuan informasi SMS yang belum terfokus menyebabkan banyaknya jawaban standar (*default replies*) masih banyak terjadi.

Fitur-fitur standar SMS Gateway, yaitu komunikasi SMS interaktif dua arah, SMS *info on demand*, SMS *service settings*, SMS *Automatic Registration*, *polling* SMS, pengiriman SMS *Broadcast*, pengiriman SMS ke *Call Group*.

Kemudian pengiriman SMS terjadwal, personalisasi SMS, antarmuka aplikasi berbasis *web*, buku alamat dan *call group*, manajemen pengguna, sistem *security access*, serta sistem parameter.

Fitur-fitur *advance SMS Gateway*, yaitu antarmuka dinamis untuk integrasi ke *database* perusahaan, *SMS Remote Control*, E-mail to SMS, SMS to E-mail, ekspansi modem GSM, dan koneksi langsung ke SMSC via SMPP [6].



**Gambar 2.8** Arsitektur SMS Gateway

SMSC (*Short Message Service Center*) merupakan jaringan telepon selular yang menangani pengiriman SMS. Jadi, pada saat seseorang mengirimkan sebuah pesan SMS melalui ponselnya, SMSC-lah yang bertugas mengirimkan pesan tersebut ke nomor tujuan. Jika nomor tujuan tidak aktif, maka SMSC akan menyimpan pesan tersebut dalam jangka waktu tertentu. Jika SMS tetap tidak dapat terkirim sampai jangka waktu tersebut berakhir, maka SMS tersebut akan dihapus dari penyimpanan SMSC.

Sebuah aplikasi SMS *gateway* dapat menggunakan jalur SMSC untuk pengoperasiannya. Keuntungannya adalah penggunaan nomor pendek yang mungkin dapat terdiri dari 3 atau 4 digit saja, misalnya 222, 9090, dan seterusnya.

Contohnya seperti saat mengikuti *polling* SMS, nomor pendek ini disediakan oleh operator jaringan SMSC. Jalur SMSC juga dapat mengirim SMS dalam jumlah banyak dalam waktu yang relatif singkat. Hanya saja, untuk membuat SMS *gateway* dengan menggunakan jalur SMSC, harus memiliki jalur koneksi ke operator selular. Umumnya layanan ini digunakan jika aplikasi dapat menghasilkan lalu lintas SMS yang tinggi.

Selain itu ada alternatif infrastruktur yang lebih sederhana dan mudah didapatkan, yaitu membuat SMS *gateway* yang menggunakan ponsel ataupun modem GSM/CDMA sebagai media pengirim/penerima SMS, di mana ponsel atau modem GSM/CDMA tersebut terpasang pada sebuah komputer.

Berikut ini peralatan yang dibutuhkan dalam membangun SMS *Gateway* adalah sebagai berikut:

- a. PC *Server*
- b. Modem GSM/CDMA
- c. *Simcard* (kartu) GSM/CDMA
- d. Modul SMS *Gateway Gammu*

## BAB III

### ANALISA DAN PERANCANGAN SISTEM

#### 3.1 Analisa Sistem

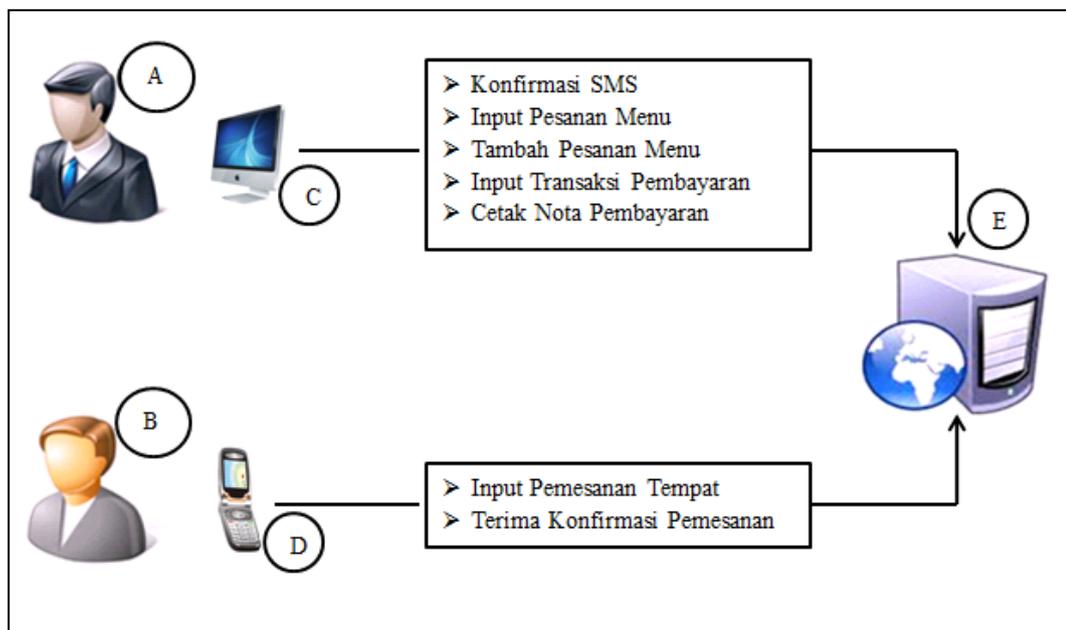
Dalam sub bab analisa sistem ini, akan dijelaskan mengenai analisa dari sistem yang dibuat, yaitu Aplikasi Pemesanan Di Resto De'kasteel. Sistem ini bertujuan untuk memberikan informasi data pemesanan, jumlah kursi pemesanan dan stok menu pada aplikasi yang dapat di gunakan untuk memudahkan transaksi pemesanan secara SMS *Gateway*.

#### 3.2 Perancangan Sistem

Perancangan sistem membahas mengenai konsep dari sistem yang akan di bangun untuk memenuhi kebutuhan kepada pengguna sistem dan memberikan gambaran secara umum dan jelas kepada *user* tentang sistem yang baru. Desain sistem secara umum merupakan persiapan dari desain dan mengidentifikasi komponen-komponen sistem yang akan didesain secara detail. Dalam melakukan perancangan terhadap sistem, maka ada dua model yang dipergunakan dalam melakukan desain sistem yaitu *process modeling* dengan menggunakan *Data Flow Diagram* (DFD) dan *data modeling* dengan menggunakan *Entity Relationship Diagram* (ERD).

##### 3.2.1 Deskripsi Umum Sistem

Secara umum, sistem yang dibuat akan ditunjukkan pada Gambar 3.1 di bawah ini:



**Gambar 3.1** Deskripsi Aplikasi Resto De'kasteel

Deskripsi dari arsitektur sistem pada Gambar 3.1 adalah sebagai berikut:

- A. *Admin* atau administrator sistem informasi menjalankan sistem informasi pemesanan online.
- B. *User* atau pengunjung sistem informasi, *user* disini dapat melakukan registrasi, melakukan transaksi pemesanan tempat meja dan terima konfirmasi pemesanan.
- C. Komputer, perangkat yang digunakan oleh admin.
- D. Handphone, perangkat yang digunakan oleh *user*.
- E. *Server* tempat proses sistem informasi yang terjadi. Jadi semua proses yang di lakukan oleh admin dan *user* tersimpan dan terjadi pada *server*.

### 3.2.2 Kebutuhan Sistem

Dengan mengidentifikasi deskripsi pada Gambar 3.1 telah diketahui bahwa terdapat beberapa fokus utama pada sistem yaitu berupa proses input meja untuk kapasitas berapa orang, proses pemesanan tempat serta proses pembayaran, semua proses data tersebut merupakan kebutuhan pengguna.

#### 3.2.2.1 Kebutuhan Pengguna

Berdasarkan arsitektur pada Gambar 3.1 untuk memenuhi kebutuhan pengguna dalam berinteraksi dengan sistem serta untuk mengetahui kebutuhan-kebutuhan apa saja yang berpengaruh pada sistem nantinya, maka perlu dijabarkan kebutuhan apa saja yang akan dibutuhkan oleh pengguna, antara lain:

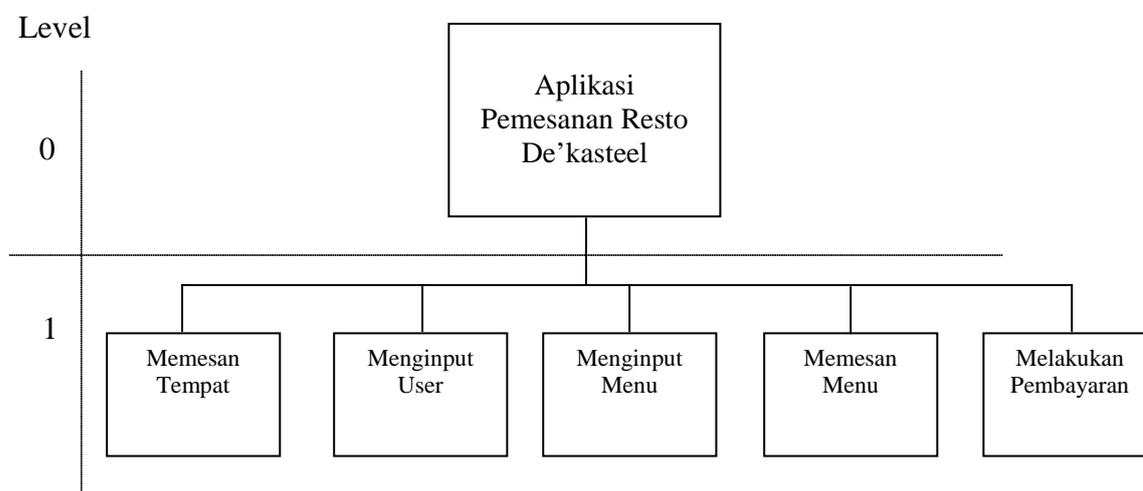
- a. *Komputer / Laptop* tidak tergantung merk, apapun bisa dipakai.
- b. Menu untuk home, sebagai halaman muka dari sistem pemesanan.
- c. Menu daftar *user*, terdapat list nama-nama *user* atau pelanggan.
- d. Menu master kasir, bertujuan untuk mengisikan isi data admin atau kasir yang mempunyai hak akses penuh dalam pengelolaan aplikasi pemesanan Resto De'kasteel.
- e. Menu master menu, di sini admin bisa menambah dan mengedit daftar menu atau produk yang akan di jual di resto de'kasteel.
- f. Menu master meja, dimana master meja bisa digunakan untuk mengisikan jumlah meja yang ada dan siap digunakan oleh *customer* yang akan melakukan pemesanan di Resto De'kasteel .
- g. Menu Daftar Meja, dimana admin bisa melihat dan sebagai info untuk mengetahui stok meja yang kosong ataupun yang terisi pada aplikasi ini.

### 3.2.3 Kebutuhan Basis Data

Dalam Aplikasi pemesanan tempat di Resto De'kasteel dibutuhkan basis data untuk mengatur semua transaksi bisnis yang terjadi, selain itu juga untuk mengatur proses transaksi pemesanan secara online.

#### 3.2.3.1 Diagram Jenjang

Hasil dari DFD yang telah dibuat diatas dapat digambarkan dengan menggunakan diagram jenjang, berikut adalah gambar diagram jenjang:



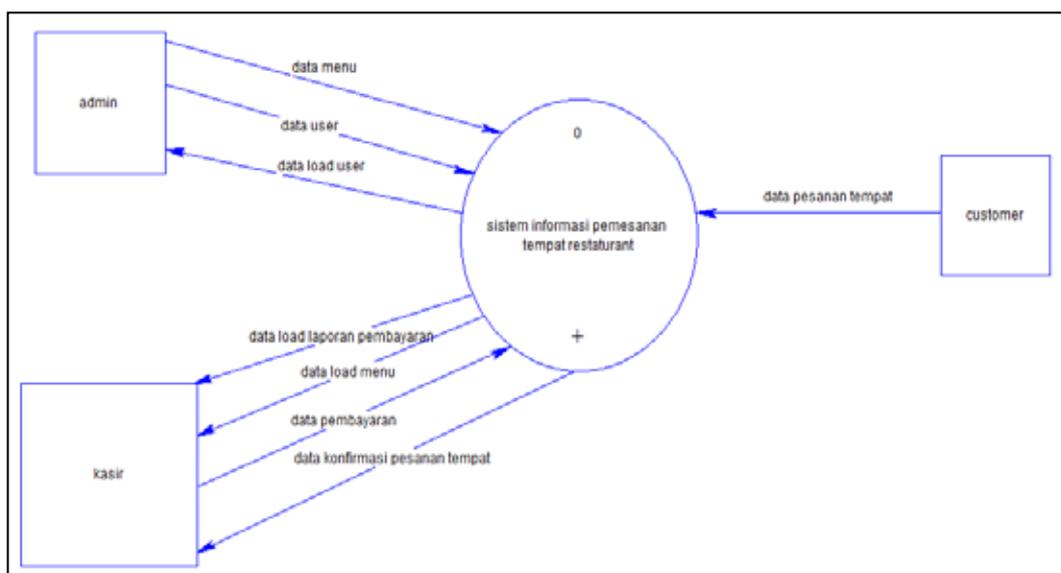
**Gambar 3.2** Diagram Jenjang

Dari diagram yang telah dibuat terdapat 2 level diantara level 0 terdapat 1 proses yang global yaitu Sistem Rekapitulasi Data dan level 1 terdapat 5 proses diantaranya adalah menginput berita, menginput produk, melakukan transaksi pemesanan, melakukan registrasi *user*, menginput guest book. Fungsi dari

diagram jenjang adalah dapat melihat gambaran dari DFD yang telah dibuat diatas tadi.

### 3.2.3.2 DFD (Data Flow Diagram)

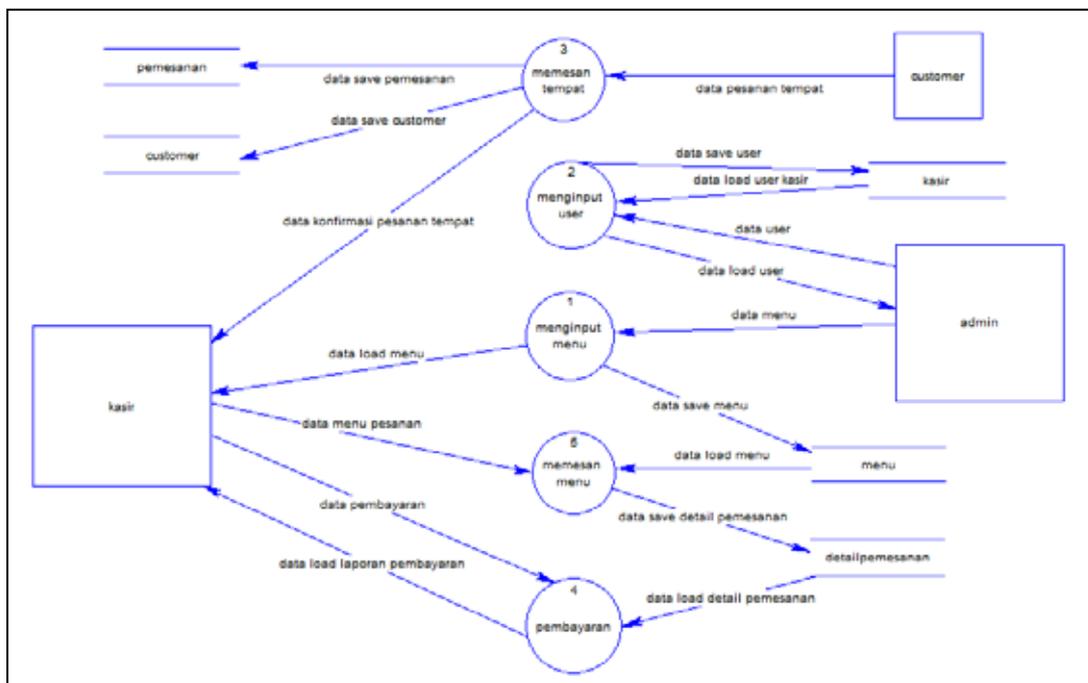
Dari Gambar 3.2. maka dapat dideskripsikan beberapa proses yang terjadi pada sistem ini. Proses tersebut dapat digambarkan sebagai berikut adalah:



**Gambar 3.3** Context Diagram

Konteks diagram pada sistem pemesanan tempat di Resto De'kasteel ini adalah gambaran garis besar dari proses yang terjadi dalam sistem tersebut. terdapat 2 entitas yaitu Admin dan User, admin dari Resto De'kasteel, dimana admin adalah seorang administrator yang memiliki hak dan tanggung jawab penuh terhadap semua proses yang terjadi pada sistem diantaranya adalah input meja, input menu, dan melakukan konfirmasi SMS, sedangkan User sendiri adalah seorang user atau pengunjung yang melakukan pemesanan tempat di Resto

De'kasteel dan dapat melakukan konfirmasi pemesanan jika ada balasan konfirmasi dari admin.



Gambar 3.4 DFD Level 1

Pada DFD level 1 ini memiliki 5 proses diantaranya adalah memesan tempat, menginput *user*, menginput menu, memesan menu dan transaksi pembayaran.

Memesan tempat adalah proses untuk melakukan pemesanan tempat di Resto De'kasteel yang dilakukan *user* pada sistem pemesanan tempat di Resto De'kasteel, kemudian pemesanan dapat di lihat oleh admin kemudian dilakukan verifikasi pemesanan.

Input *user* dapat dideskripsikan sebagai berikut, *user* yang di maksud adalah admin, di mana admin ini mempunyai hak akses penuh pada Aplikasi

pemesanan tempat. Admin melakukan konfirmasi sms, input menu, dan melakukan pembayaran.

Input menu dilakukan oleh admin. Admin melakukan pengisian data produk menu yang dijual oleh Resto De'kasteel. Pemesanan menu dilakukan oleh admin. Admin melakukan transaksi pemesanan menu yang dilakukan oleh pembeli sesuai apa yang di pesan oleh *customer*.

Pembayaran ini dilakukan oleh admin. *user* melakukan pembayaran atas menu yang telah di pesan oleh *customer*.

### **3.2.4 ERD (*Entity Relational Diagram*)**

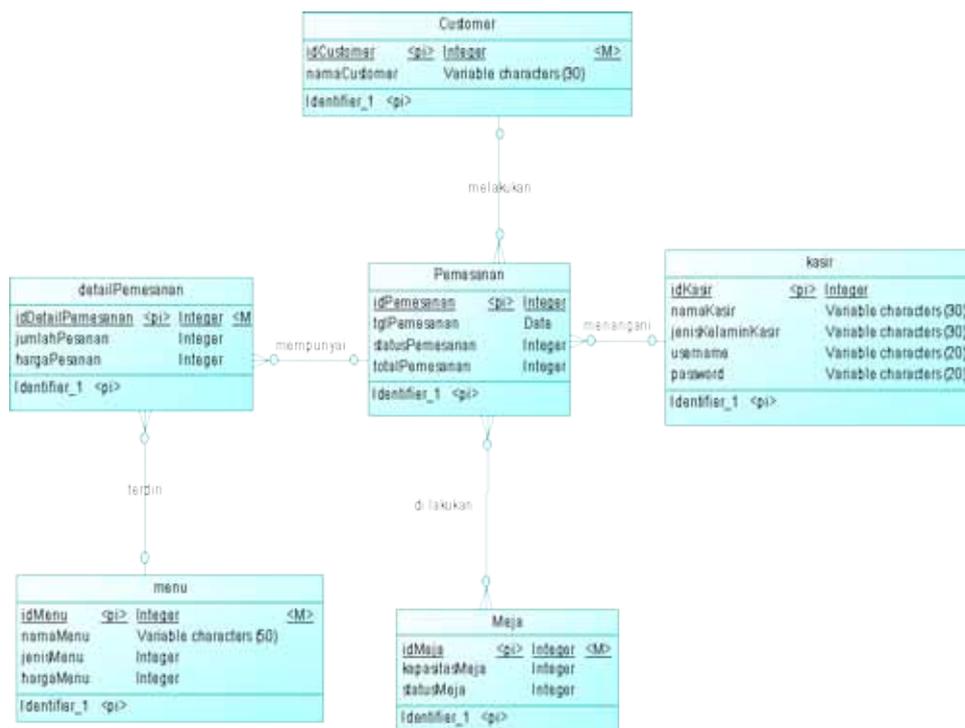
Setelah mendesain DFD, kemudian mendesain ERD dimana ERD ini nantinya akan dibuat *relational* tabelnya. ERD mendokumentasikan data dengan mengidentifikasi tipe dari data entitas. *Entity Relationship Diagram* merupakan *data modeling* yang dipergunakan untuk mendokumentasikan data sebuah perusahaan dengan cara menentukan data apa saja yang terdapat dalam tiap *entity* dan bagaimana hubungan antara *entity* satu dengan lainnya.

*Entity Relational Diagram* terbagi menjadi 2, yaitu *Conceptual Data Model* dan *Physical Data Model*. *Conceptual Data Model* adalah rancangan dalam bentuk diagram *logic* tanpa mempertimbangkan DBMS. *Physical Data Model* adalah rancangan fisik basis data yang disimpan di *server*.

#### **3.2.4.1 CDM (*Conceptual Data Model*)**

*Conceptual Data Model* adalah rancangan dalam bentuk diagram sebelum pembuatan database secara detail. Model data konseptual ini dibuat tanpa harus mempertimbangkan DBMS apa yang nantinya dipakai, karena bentuk ini hanya sebuah konsep rancangan yang nantinya bisa digunakan oleh semua DBMS.

*Conceptual Data Model* pada aplikasi ini mempresentasikan rancangan bisnis data konseptual yang nantinya akan digunakan dalam sistem ini. Pada konsep rancangan ini dibuat dengan menggunakan Power Designer 12. untuk lebih jelasnya dapat dilihat pada Gambar 3.4 berikut:

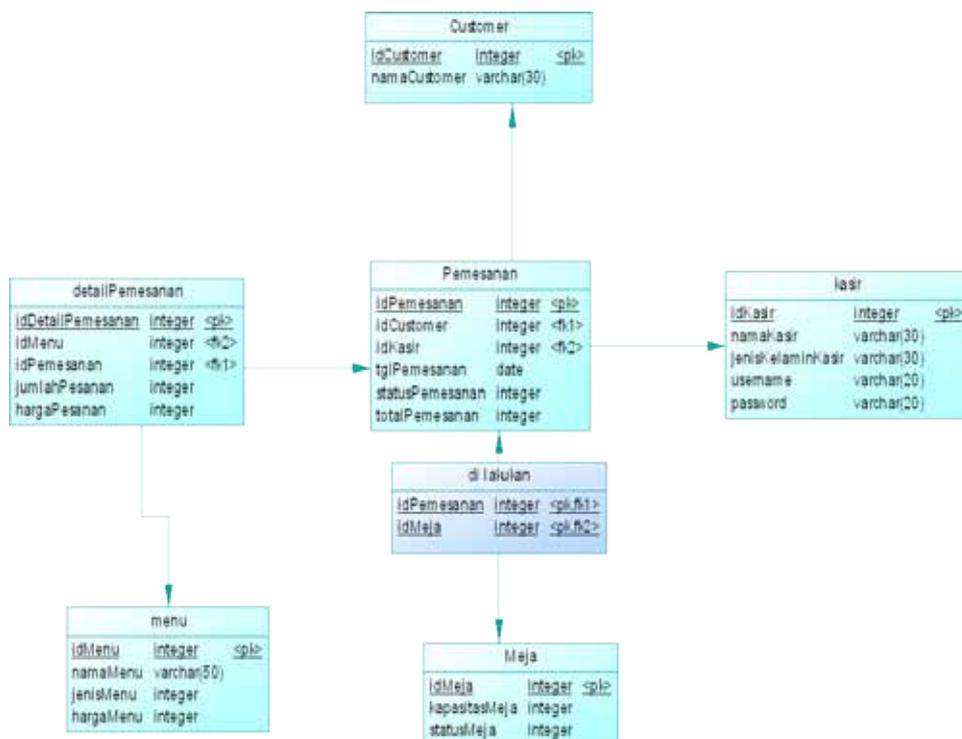


**Gambar 3.5** CDM (*Conceptual Data Model*)

### 3.2.4.2 PDM (*Physical Data Model*)

PDM (*Physical Data Model*) adalah sebuah model struktur fisik dari sebuah *database* yang mempertimbangkan DBMS dan struktur data penyimpanan. PDM merupakan representasi fisik dari *database* yang akan dibuat dengan mempertimbangkan DBMS (*Database Management System*) yang akan digunakan. *Physical data model* merupakan model yang menggunakan sejumlah tabel untuk menggambarkan data serta hubungan antara data-data tersebut. Setiap

tabel mempunyai sejumlah kolom di mana setiap kolom memiliki nama yang unik. Pada Gambar 3.5 di bawah ini ditunjukkan gambar dari *Physical Data Model* dari sistem informasi penggajian dan pengelolaan pegawai:



**Gambar 3.6** PDM (*Physical Data Model*)

### 3.2.5 Database Relational

Model relasional berdasarkan teori himpunan matematik. Struktur didefinisikan dengan tabel. Dalam istilah matematika, tabel disebut sebagai Relasi. Tiap tabel dalam model relasional dikomposisikan dari baris dan kolom, kolom disebut Atribut. Nilai untuk sebuah atribut harus dipilih dari sekelompok nilai yang dinamakan domain. Karena banyak kolom dalam tabel yang sama dapat diidentifikasi atas domain yang sama, maka nama atribut didefinisikan untuk

tiap kolom dan tiap nama atribut dalam sebuah relasi harus unik. Di bawah ini adalah sifat-sifat dari tabel :

- a. Duplikasi baris tidak diperbolehkan. Untuk melaksanakan sifat ini, harus terdapat paling sedikit satu atribut atau kombinasi beberapa atribut yang mengidentifikasi secara unik tiap baris dari tabel. Atribut atau kombinasi beberapa atribut yang melaksanakan tugas ini disebut **Kunci Primer** (*Primary Key*). Contoh: *idUser*, adalah kunci primer yang mengidentifikasi tiap *user* secara unik.
- b. Database relasional adalah nilai *Primary Key* tidak boleh mempunyai duplikat atau NIL (*NULL*, yaitu nilai tidak diketahui).
- c. Keterhubungan (*relationship*) antara dua tabel. Jika Tabel R2 mempunyai sebuah **Kunci Asing** (*Foreign Key*) yang cocok dengan kunci primer dari Tabel R1, maka untuk setiap nilai *Foreign Key* harus terdapat sebuah nilai kecocokan dari *Primary Key*, atau nilai *Foreign Key* harus nil.

Berikut ini adalah *database relational* berdasarkan struktur tabel yang telah di *generate* dari *physical data model* yang telah dibuat:

**a. Tabel Customer**

Tabel berita digunakan untuk menyimpan data *customer* dari resto De'Kasteel, berikut adalah tabel customer:

*Primary Key* : *idCustomer*

	Name	Code	Data Type	Length	Precision	P	F	M
1	idCustomer	IDCUSTOMER	integer			<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
2	namaCustomer	NAMACUSTOM	varchar(30)	30		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

**Gambar 3.7** Tabel *Customer***b. Tabel Pemesanan**

Tabel pemesanan digunakan untuk menyimpan data pemesanan di Aplikasi Pemesanan Online Di Resto De'kasteel, berikut adalah tabel komentar :

*Primary Key* : idPemesanan

*Foreign Key* : idMeja

*Foreign Key* : idCustomer

*Foreign Key* : idKasir

	Name	Code	Data Type	Length	Precision	P	F	M
1	idPemesanan	IDPEMESANAN	integer			<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
2	idMeja	IDMEJA	integer			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
3	idCustomer	IDCUSTOMER	integer			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
4	idKasir	IDKASIR	integer			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
5	tglPemesanan	TGLPEMESAN	date			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
6	statusPemesanan	STATUSPEME	integer			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
7	totalPemesanan	TOTALPEMESA	integer			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

**Gambar 3.8** Tabel Pemesanan**c. Tabel Detail Pemesanan**

Tabel komentar digunakan untuk menyimpan data detail pemesanan menu pada Aplikasi Pemesanan Online Di Resto De'kasteel, berikut adalah tabel *user* :

*Primary Key* : idDetailPemesanan

*Foreign Key* : idMenu

*Foreign Key* : idPemesanan

	Name	Code	Data Type	Length	Precision	P	F	M
1	idDetailPemesanan	IDDETAILPEME	integer			<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
2	idMenu	IDMENU	integer			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
3	idPemesanan	IDPEMESANAN	integer			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
4	jumlahPesanan	JUMLAHPEMAN	integer			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
5	hargaPesanan	HARGAPEMAN	integer			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

**Gambar 3.9** Gambar Tabel Detail Pemesanan

#### d. Tabel Kasir

Tabel transaksi digunakan untuk menyimpan data kasir di Resto De'Kasteel, berikut adalah tabel kasir :

*Primary Key* : idKasir

	Name	Code	Data Type	Length	Precision	P	F	M
1	idKasir	IDKASIR	integer			<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
2	namaKasir	NAMAKASIR	varchar(30)	30		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
3	jenisKelaminKasir	JENISKELAMIN	varchar(10)	10		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
4	username	USERNAME	varchar(20)	20		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
5	password	PASSWORD	varchar(20)	20		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

**Gambar 3.10** Tabel Kasir

#### e. Tabel Meja

Tabel meja digunakan untuk menyimpan data meja yang dapat menampung kapasitas *customer*, berikut adalah tabel detail meja :

*Primary Key* : idMeja

	Name	Code	Data Type	Length	Precision	P	F	M
1	idMeja	IDMEJA	integer			<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
2	kapasitasMeja	KAPASITASME	integer			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
3	statusMeja	STATUSMEJA	integer			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

**Gambar 3.11** Tabel Meja

#### f. Tabel Menu

Tabel produk digunakan untuk menyimpan data produk menu yang ada di resto De'Kasteel, berikut adalah tabel produk :

*Primary Key* : idMenu

	Name	Code	Data Type	Length	Precision	P	F	M
1	idMenu	IDMENU	integer			<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
2	namaMenu	NAMAMENU	varchar(50)	50		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
3	jenisMenu	JENISMENU	integer			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
4	hargaMenu	HARGAMENU	integer			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

**Gambar 3.12** Tabel Menu

### 3.3 Perancangan Antar Muka (*Interface*)

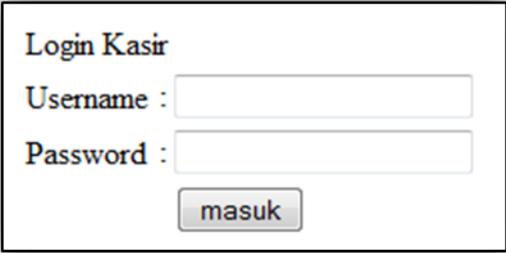
Perancangan antarmuka merupakan perancangan halaman aplikasi yang berinteraksi langsung antara sistem dengan pengguna. Perancangan antarmuka sistem merupakan salah satu bagian penting dalam tahap perancangan sistem. Dalam merancang antar muka aplikasi harus memudahkan pengguna dalam mengoperasikan sistem atau aplikasi tersebut.

Konsep rancangan yang sangat penting untuk mendesain halaman *web* adalah resolusi monitor. Terdapat 3 buah resolusi yang umum digunakan pada monitor, diantaranya 640x480, 800x600, 1024x768 pixel. Pada umumnya halaman *web* didesain untuk resolusi 800x600 karena pada umumnya pengguna sistem memakai monitor ukuran 14 atau 15 inchi.

Dalam merancang tampilan sebuah situs, harus menggambarkan *layout* dari situs yang akan dibuat. *Layout* berfungsi untuk memudahkan dalam mendesain tampilan pada sebuah halaman *web*.

### 3.3.1 Perancangan Antar Muka Halaman Pembuka

Perancangan halaman utama dari Pemesanan tempat pada Resto De'kasteel berbasis SMS *gateway*, akan menampilkan menu *login*. Halaman ini adalah *login* admin sebelum masuk atau *login* untuk dapat mengakses ke halaman selanjutnya.



The image shows a login form with the following elements:

- Title: Login Kasir
- Username field: Username :
- Password field: Password :
- Submit button: masuk

**Gambar 3.13** Form Halaman Login

### 3.3.2 Perancangan Antar Muka Halaman Utama Admin

Perancangan halaman utama dari aplikasi pemesanan Resto De'kasteel berbasis website, akan menampilkan data data informasi tentang aplikasi pemesanan tempat di Resto De'kasteel, yaitu input menu, tambah *user* kasir, dan tambah meja. Pada Gambar 3.13 di bawah ini adalah gambar untuk input menu, yang berguna untuk memasukan menu pemesanan dari *customer*.



The image shows a web form titled "Tambah Menu". It contains three input fields: "Nama Menu" (text input), "Jenis Menu" (dropdown menu), and "Harga" (text input). Below the inputs is a "simpan" button.

**Gambar 3.14** Interface Halaman Login

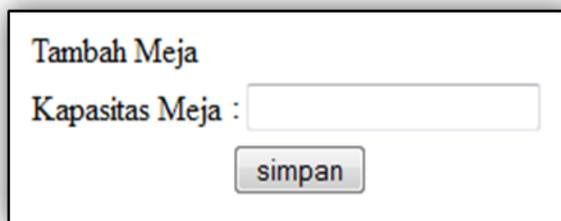
Pada tambah *user*, berfungsi untuk menambah dan menyimpan data Admin atau kasir yang berhak penuh dalam aplikasi pemesanan, berikut Gambar 3.14 adalah gambar interface *user* kasir.



The image shows a web form titled "Tambah User Kasir". It contains four input fields: "Nama" (text input), "Jenis Kelamin" (dropdown menu), "Username" (text input), and "Password" (text input). Below the inputs is a "simpan" button.

**Gambar 3.15** Interface Halaman Login

Pada tambah meja, berfungsi untuk menambah jumlah pemesanan meja yang telah di pesan oleh *customer*, berikut Gambar 3.15 adalah gambar interface tambah meja.

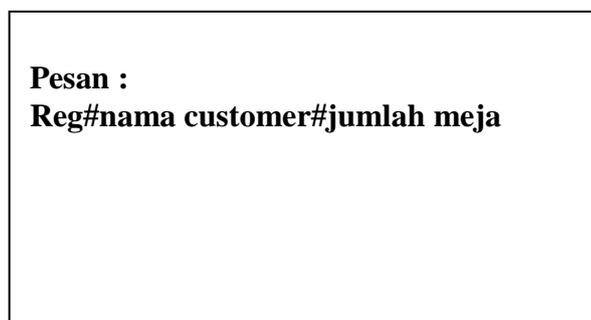


Tambah Meja  
Kapasitas Meja :

**Gambar 3.16** Interface Halaman Tambah Meja

### 3.4 Perancangan Format SMS

Perancangan format SMS yang di gunakan untuk melakukan pemesanan yaitu mengetikan REG#NAMA#JUMLAHMEJA. Sehingga dengan adanya sistem SMS gateway memudahkan *customer* dalam melakukan pemesanan



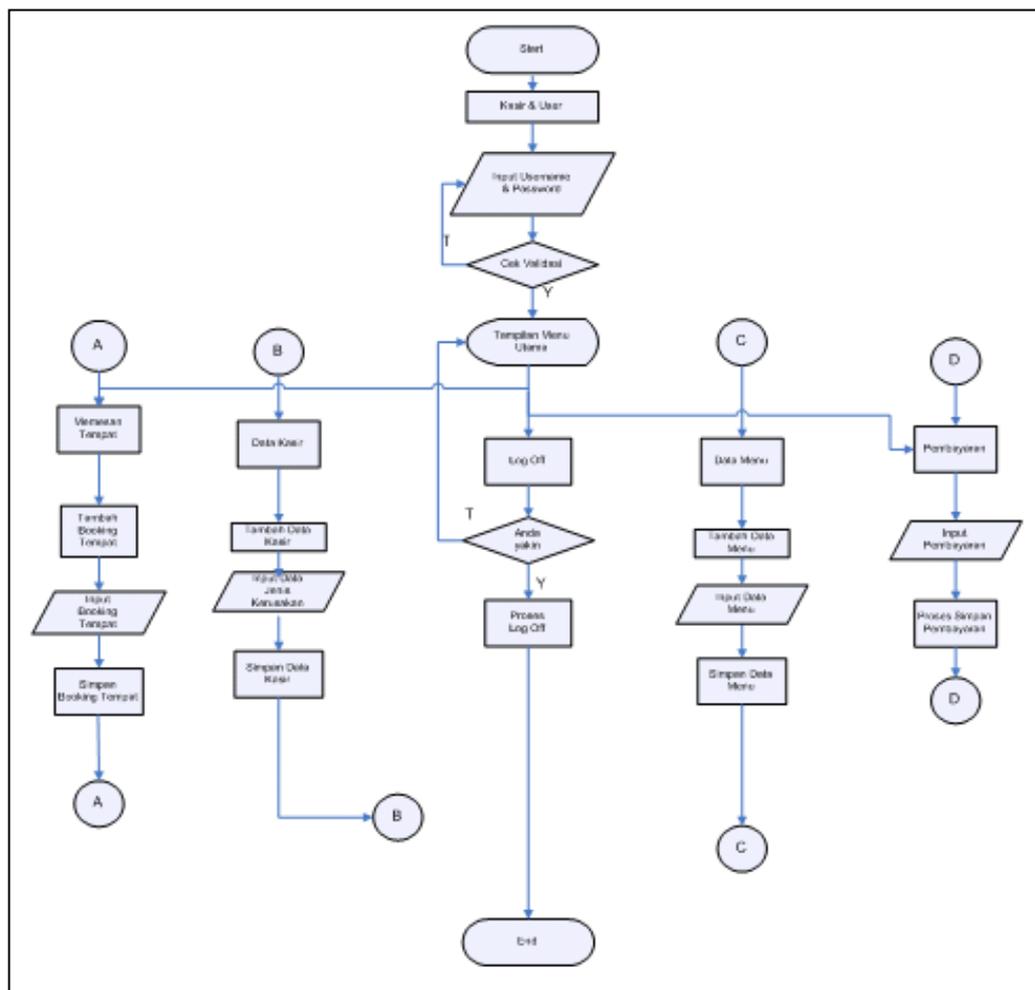
**Pesan :**  
**Reg#nama customer#jumlah meja**

**Gambar 3.17** Format Pemesanan

### 3.5 Flowchart

Yaitu bagai mana alur untuk memasukkan data ke sistem sehingga data dapat tersimpan dengan sebagai mana mestinya. Terdapat urutan pada saat pemakaian aplikasi dalam hal pengolahan data dengan aplikasi ini.

Bagan alir skematik (schematic flowchart) merupakan bagan alir yang mirip dengan bagan alir sistem, yaitu untuk menggambarkan prosedur di dalam sistem. Perbedaannya adalah, bagan alir skematik selain menggunakan simbol-simbol bagan alir sistem, juga menggunakan gambar-gambar komputer dan peralatan lainnya yang digunakan. Maksud penggunaan gambar-gambar ini adalah untuk memudahkan komunikasi kepada orang yang kurang paham dengan simbol-simbol bagan alir. Penggunaan gambar-gambar ini memudahkan untuk dipahami, tetapi sulit dan lama menggambarinya.



Gambar 3.1 *Flowchart Aplikasi Pemesanan Online Berbasis SMS Gateway Studi Kasus di Resto De' Kasteel*

## **BAB IV**

### **IMPLEMENTASI SISTEM**

Pada bab IV ini akan dibahas mengenai implementasi dari rancangan sistem yang telah dibuat pada bab III. Bagian implementasi sistem kali ini meliputi: lingkungan implementasi, implementasi proses, dan implementasi antarmuka.

#### **4.1 Lingkungan Implementasi**

Pada bagian ini akan dijelaskan mengenai perangkat keras dan perangkat lunak yang digunakan pada implementasi sistem ini.

Perangkat keras:

- Notebook Toshiba dengan processor AMD Turion X2
- Memori sodimm 3 GB.
- Modem untuk koneksi data.

Perangkat lunak:

- Sistem Operasi Windows Vista
- Dreamweaver 8 untuk mendesain aplikasi ini
- Php My admin (*localhost*)
- MySQL (*database* aplikasi)
- Web browser menggunakan Mozilla Firefox

## 4.2. Implementasi

Pada sub bab Implementasi aplikasi desain antarmuka menjelaskan apa saja yang terlibat dalam aplikasi yang telah dibuat berdasarkan perancangan yang telah di bahas pada bab III. Pada sistem ini terdapat beberapa menu utama dari Aplikasi Resto De'kasteel antara lain yaitu:

- a. Halaman tampilan login
- b. Halaman Pembuka
- c. Halaman Master Kasir
- d. Halaman Master Menu
- e. Halaman Master Meja
- f. Halaman Daftar Meja
- g. Implementasi SMS Gateway

### 4.2.1 Form Tampilan Login

Pada halaman login aplikasi dari Aplikasi pemesanan Resto De'kasteel, user diwajibkan login terlebih dahulu, untuk mendapatkan hak akses. Berikut adalah tampilan login dari aplikasi :



The image shows a login interface for a restaurant. At the top, there is a light green header with the word "Restaurant" in a serif font. Below this, the text "Selamat Datang di Restaurant" is displayed in a white serif font. Underneath, the word "Login" is shown in a smaller white font. The login form is enclosed in a dark grey box and contains two input fields: "Login Kasir" (with sub-labels "Username" and "Password") and a "masuk" button.

**Gambar 4.1** Halaman Login

#### 4.2.2 Halaman Pembuka

Halaman pembuka ini adalah halaman utama atau pembuka setelah user atau kasir berhasil login dan mendapatkan hak akses pada aplikasi pemesanan Resto De'kasteel. Berikut adalah tampilan dari halaman pembuka di Resto De'kasteel:



**Gambar 4.2** Halaman Pembuka

#### 4.2.3 Halaman Master Kasir

Halaman master kasir digunakan untuk melakukan tambah atau edit data admin atau kasir yang nantinya mempunyai hak akses penuh dalam melakukan pengelolaan aplikasi pemesanan Resto De'kastell. Berikut adalah gambar halaman master kasir:



**Gambar 4.3** Halaman Mater Kasir

#### 4.2.4 Halaman Master Menu

Halaman master menu ini di gunakan untuk menambah atau mengedit daftar menu yang ada pada resto de'kasteel. Nantinya menu tersebut bisa di gunakan sebagai produk yang akan di jual kepada customer. Berikut adalah gambar Halaman Master Menu:

No.	Nama Menu	Jenis Menu	Harga Menu	
1.	Spaghetti	Makanan	16000	edit
2.	nasi goreng ayam manis	Makanan	25000	edit

Gambar 4.4 Halaman Master Menu

#### 4.2.5 Halaman Master Meja

Pada halaman master meja di gunakan untuk mengisikan daftar meja yang ada pada Resto De'kasteel. Dimana meja tersebut dapat di gunakan admin untuk mengetahui jumlah meja yang kosong dan terisi. Berikut adalah isi dari halaman master meja :

No. Meja	Kapasitas	Status
1.	2 Orang	Terisi
2.	1 Orang	Kosong

Gambar 4.5 Halaman Master Meja

#### 4.2.6 Halaman Daftar Meja

Pada halaman daftar meja di gunakan untuk melihat data meja yang sedang terisi atau kosong, kalau yang terisi, ada pilihan menu dan bayar, menu di gunakan untuk input pemesanan menu sedangkan bayar untuk melakukan pembayaran. Berikut adalah tampilanya:

No.	Kapasitas	Status	
1.	2 Orang	Terisi	menu Bayar
2.	1 Orang	Kosong	Pesan

**Gambar 4.6** Halaman Daftar Meja

#### 4.2.7 Implementasi SMS Gateway

Implemtasi ini di gunakan untuk bagaimana format penulisan sms customer yang akan di kirimkan untuk dapat melakukan pemesanan tempat meja

**Pesan :**  
**Reg#nama customer#jumlah meja**

**Gambar 4.7** Format Pemesanan

Setelah customer mengirimkan pesan singkat seperti pada contoh format di atas, maka customer hanya dapat 2 jenis balasan, balasan ketika pemesanan berhasil, dan balasan ketika tempat sudah penuh.

**Pesan :**  
**TERIMA KASIH, PEMESANAN**  
**ANDA BERHASIL DISIMPAN,**  
**DITUNGGU KEDATANGAN**  
**ANDA DI RESTAURANT**  
**D'KASTEEL**

**Gambar 4.8** Konfirmasi Berhasil

## **BAB V**

### **UJI COBA DAN EVALUASI**

Pada bab ini akan membahas tentang uji coba dan evaluasi program yang menerangkan bagaimana jalanya program secara detail yang akan di jelaskan dibawah ini:

#### **5.1 Uji Coba**

Uji coba bertujuan untuk memastikan bahwa aplikasi telah dibuat dengan benar sesuai dengan kebutuhan atau tujuan yang diharapkan.

##### **5.1.1 Uji Coba Sistem**

Adapun uji coba didalam aplikasi pemesanan Resto De'kasteel ini yaitu uji coba dari semua menu yang ada.

Untuk memeriksa sistem berjalan dengan lancar dan sesuai dengan proses yang telah dijelaskan pada bab-bab sebelumnya maka dibuat skenario yang terdiri dari beberapa uji coba diantaranya :

- a. Uji Coba Login
- b. Uji Coba Input Data Kasir
- c. Uji Coba Input Tambah Menu
- d. Uji Coba Input Tambah Meja
- e. Uji Coba SMS Gateway

## 5.2 Uji Coba Login

Pada uji coba ini akan di uji cobakan dalam melakukan login pada aplikasi pemesanan Resto De'kasteel.

- a. Pada Gambar 5.1 dibawah ini menunjukkan halaman login, form username dan password harus di isi sesuai yang ada data yang valid.



**Gambar 5.1** Gambar Login

- b. Jika login berhasil, maka akan langsung masuk ke dalam aplikasi, seperti pada Gambar 5.2 di bawah akan langsung menunjukkan halaman utama dari aplikasi.



**Gambar 5.2** Halaman Admin

### 5.3 Uji Coba Input Data Kasir

Pada uji coba ini akan di uji cobakan untuk input data kasir

- a. Pada Gambar 5.3 adalah tampilan dari gambar form input data kasir, jika ingin mengakses di wajibkan untuk registrasi terlebih dahulu.

The screenshot shows a web form titled "Tambah Kasir". It has a sub-header "Tambah User Kasir". The form contains four input fields: "Nama" with the value "Raden Bejo", "Jenis Kelamin" with a dropdown menu showing "Laki-laki", "Username" with the value "bejo", and "Password" with masked characters "\*\*\*\*\*". A "simpan" button is located at the bottom of the form.

**Gambar 5.3** Halaman Tambah Kasir

- b. Setelah menginputkan data, data akan tersimpan di database dan akan terlihat pada aplikasi, pada Gambar 5.4 adalah tampilan dari halaman daftar nama kasir yang tersimpan.

No.	Nama Kasir	Jk	Username	Password	
1.	Ratna	Perempuan	ratna	123	<a href="#">edit</a>
2.	Galih	Laki-laki	galih	1234	<a href="#">edit</a>
3.	iis	Perempuan	iis	mamaku	<a href="#">edit</a>
4.	Raden Bejo	Laki-laki	bejo	upn	<a href="#">edit</a>

**Gambar 5.4** Halaman Daftar Kasir

#### 5.4 Uji Coba Input Tambah Menu

Pada uji coba ini akan di uji cobakan untuk input tambah menu.

- a. Pada Gambar 5.5 adalah tampilan dari gambar tambah menu, jika ingin menambah daftar menu klik menu master menu.



**Gambar 5.5** Halaman Tambah Menu

- b. Setelah data terinputkan, akan terlihat pada daftar menu yang ada pada aplikasi, pada Gambar 5.6 adalah tampilan halaman daftar menu.

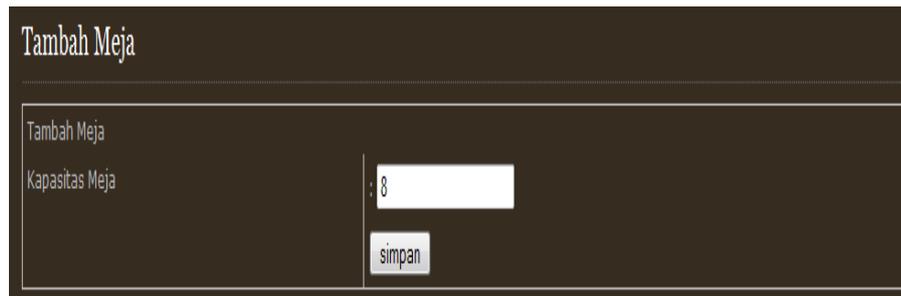
No	Nama Menu	Jenis Menu	Harga Menu	
1.	Spaghetti	Makanan	11000	edit
2.	bakso	Makanan	7000	edit
3.	Crispy Chicken With Fried Rice	Makanan	17000	edit

**Gambar 5.6** Halaman Daftar Menu

### 5.5 Uji Coba Input Tambah Meja

Pada uji coba ini akan di uji cobakan untuk input produk.

- a. Pada Gambar 5.8 adalah tampilan dari gambar halaman tambah meja, jika ingin masuk klik master meja.



**Gambar 5.7** Halaman Tambah Meja

- b. Setelah di inputkan, akan terlihat pada halaman daftar meja, pada Gambar 5.9 adalah tampilan halaman daftar meja.

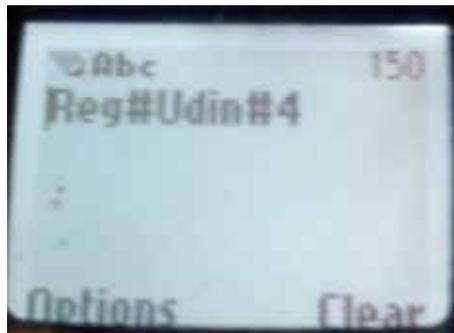
No. Meja	Kapasitas	Status	
1.	2 Orang	Kosong	
2.	4 Orang	Kosong	
3.	2 Orang	Kosong	
4.	2 Orang	Kosong	
5.	4 Orang	Kosong	
6.	6 Orang	Kosong	
7.	4 Orang	Kosong	
8.	8 Orang	Kosong	

**Gambar 5.8** Halaman Daftar Meja

## 5.6 Uji Coba Transaksi SMS Gateway

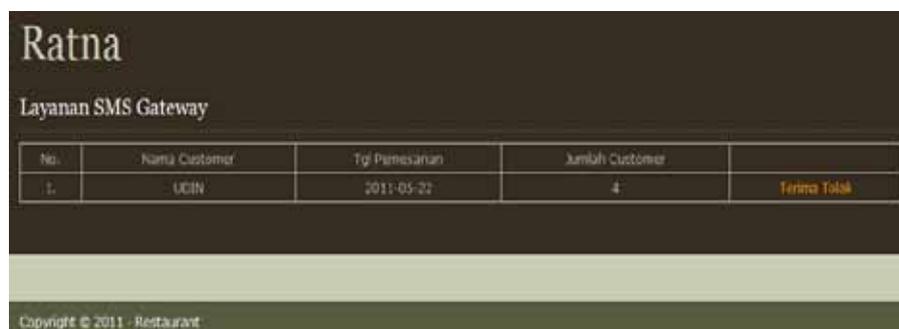
Pada uji coba ini akan di uji cobakan untuk transaksi pemesanan dengan menggunakan SMS Gateway.

- a. Pertama-tama customer mengirimkan pesan singkat ke Resto De'kasteel seperti pada gambar di bawah ini.



**Gambar 5.9** Contoh Format SMS

- b. Setelah itu format SMS tertampung pada aplikasi pemesanan Resto De'kasteel.



**Gambar 5.10** Halaman Konfirmasi Transaksi

- c. Setelah itu admin klik menu terima yang ada pada aplikasi, selanjutnya akan diproses dan klik button kirim seperti pada gambar dibawah.



**Gambar 5.11** Halaman Konfirmasi Transaksi

- d. Setelah itu customer akan mendapat balasan SMS yang isinya adalah balasan dari Resto De'kasteel seperti pada gambar di bawah ini.



**Gambar 5.12** Format Konfirmasi

## **BAB VI**

### **PENUTUP**

#### **6.1. Kesimpulan**

Setelah melakukan uji coba maka dapat diambil kesimpulan yaitu :

- a. Cara pengunjung dapat melakukan transaksi pemesanan secara online untuk membuat sistem yang terkomputerisasi dengan menggunakan beberapa komponen. Diantaranya, komponen input, komponen model, komponen output, komponen teknologi, komponen hardware, komponen software, dan komponen basis data dengan menambahkan sistem SMS Gateway.
- b. Caranya chef dapat melihat daftar customer untuk membuat sistem yang dapat memberikan informasi dengan memanfaatkan database sebagai penyedia tempat penyimpanan data, sehingga data-data yang masuk dan tersimpan bisa sebagai wadah informasi.

#### **6.2 Saran**

Berdasarkan pengalaman dalam proses pembuatan aplikasi ini, terdapat beberapa saran yang diusulkan oleh penyusun terkait untuk pengembangan aplikasi lebih lanjut. Saran-saran tersebut antara lain:

- a. Aplikasi ini bisa di kembangkan dalam bentuk mobile yang dapat menghubungkan aplikasi pemesanan, sehingga customer bisa mengases melalui jaringan internet.

## DAFTAR PUSTAKA

Eko Heri Susanto. 2008. **Membangun Aplikasi Web Dengan Ajax.**

Dari <http://www.jawadwipa.com>, diakses *online* 08-10-2010

Hakim, Lukman. 2008. **Membongkar Trik Rahasia Para Master PHP.**

Lokomedia. Yogyakarta.

Kurniawan, Rulianto. 2007. **54 Trik Tersembunyi PHP.** Maxicom. Palembang.

Prasetyo, Didik Dwi. 2006. **101 Tip & Trik Pemrograman PHP.** Elex Media

Komputindo. Jakarta.

Rafiza. 2006. **Panduan dan Referensi Kamus Fungsi PHP 5.** PT Elex Media

Komputindo. Jakarta.

## DAFTAR PUSTAKA

- Eko Heri Susanto. 2008. **Membangun Aplikasi Web Dengan Ajax**. Dari <http://www.jawadwipa.com> , diakses online 08-10-2010
- Hakim, Lukman. 2008 **Membongkar Trik Rahasia Para Master PHP**. Lokomedia. Yogyakarta.
- Kurniawan, Rulianto. 2007 . **54 Trik Tersembunyi PHP**. Maxicom. Palembang
- Prasetyo, Didik Dwi. 2006. **101 Tip & Trik Pemrograman PHP**. Elex Media Komputindo. Jakarta.
- Rafiza. 2006. **Panduan dan Referensi Kamus Fungsi PHP 5**. PT Elex Media Komputindo. Jakarta