

**PERANCANGAN DAN IMPLEMENTASI
SISTEM INFORMASI *E-MARKETPLACE*
UNTUK KATERING**

SKRIPSI

Diajukan sebagai salah satu syarat untuk memperoleh gelar kesarjanaan
Pada Program Studi Teknik Informatika
Jenjang Pendidikan Strata-1



Disusun oleh :

**MOHAMMAD ANDREAN EKA SAPUTRA
NPM : 121100019**

**FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS RESPATI INDONESIA
JAKARTA
2017**

PERNYATAAN ORISINALITAS

Yang bertanda tangan dibawah ini, saya :

Nama lengkap : MohammadAndreanEkaSaputra
NPM : 121100019
Program Studi : Teknik Informatika
Tahun akademik : 2012/2013

Menyatakan bahwa Skripsi ini adalah hasil karya saya sendiri, dan semua sumber baik yang dikutip maupun dirujuk telah saya nyatakan dengan benar. Saya tidak melakukan kegiatan plagiat dalam skripsi saya yang berjudul :

Perancangan Dan Implementasi Sistem Informasi E-Marketplace Untuk Katering

Apabila suatu saat nanti terbukti saya melakukan plagiat, maka saya akan menerima sanksi yang telah ditetapkan.

Demikian surat pernyataan ini saya buat dengan sebenar-benarnya.

Jakarta, 16 Juli 2017

Saya yang menyatakan,

Materai Rp.6000

MohammadAndreanEkaSaputra
NPM : 121100019

PERSETUJUAN

PERANCANGAN DAN IMPLEMENTASI SISTEM INFORMASI
E-MARKETPLACE UNTUK KATERING

SKRIPSI

Disusun oleh :

MohammadAndreanEkaSaputra
NPM : 121100019

Disetujui oleh :
Dosen Pembimbing

Tony Sugiarto, SE, M.Kom

Mengetahui :

Dekan
Fakultas Teknologi Infomasi

Ketua Program Studi
Teknik Informatika

Drs.Djanuri, MM

Andi Susilo, S.Kom, M.T.I

PENGESAHAN

PERANCANGAN DAN IMPLEMENTASI SISTEM INFORMASI E-MARKETPLACE UNTUK KATERING

SKRIPSI

Disusun oleh :

MohammadAndreanEkaSaputra
NPM : 121100019

Telah dipertahankan pada sidang skripsi jenjang S1 Program Studi Teknik Informatika, Fakultas Teknologi Informasi, Universitas Respati Indonesia, Pada hari Senin tanggal 7 Agustus 2017.

No.	Panitia Sidang		Nama Lengkap	Tanda Tangan
1.	Ketua	:	AndiSusilo, S.Kom, M.T.I	
2.	Sekretaris	:	Suwarni, S.Kom, M.SI	
3.	Dosen Penguji I	:	Suwarni, S.Kom, M.SI	
4.	Dosen Penguji II	:	Desmiwati, S.Kom, M.SI	
5.	Moderator	:	Tata Sutabri, S.Kom, M.SSI	

PERSETUJUAN DEMO PROGRAM

NPM : 121100019
Nama : MohammadAndreasEkaSaputra
Fakultas : Teknologi Informasi
Program Studi : Teknik Informatika
Jenjang : Strata Satu (S1)
Judul Skripsi : **Perancangan Dan Implementasi Sistem Informasi *E-Marketplace* Untuk Katering.**

Program telah disetujui dan didemokan sesuai dengan judul Skripsi diatas di hadapan Dosen Pembimbing Fakultas Teknologi Informasi Universitas Respati Indonesia, Program Studi Teknik Informatika pada tanggal:
7 Agustus 2017.

Mengetahui :

Ketua Program Studi
Teknik Informatika

(Andi Susilo, S.Kom, M.T.I)

Disetujui Oleh :

Dosen Pembimbing

(Tony Sugiarto, SE, M. Kom)



**KONSULTASI BIMBINGAN
SKRIPSI**

**FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS RESPATI INDONESIA**

NPM	: 121100019
Nama	: Mohammad Andrean Eka Saputra
Program Studi	: Teknik Informatika
Judul Skripsi	: Perancangan dan Implementasi Sistem Informasi <i>E-Marketplace</i> Untuk Katering
Dosen Pembimbing Materi	: Tony Sugiarto, SE, M.Kom

No.	Tanggal Bimbingan	Pokok Bahasan	Paraf Pembimbing
1.			
2.			
3.			

4.			
5.			
6.			
7.			
8.			

Jakarta, 16 Juli 2017

Disetujui oleh
Dosen Pembimbing Materi

(Tony Sugiarso, SE, M.Kom)

PERANCANGAN DAN IMPLEMENTASI SISTEM INFORMASI

E-MARKETPLACE UNTUK KATERING

Mohammad Andrean Eka Saputra
Program Studi Teknik Informatika
Alamat : Jl. Bambu Apus 1 No. 3
Cipayung Jakarta Timur
E-mail : andreanekasaputra@gmail.com

ABSTRAK

Aktivitas manusia yang terus meningkat dan telah menyita waktu menuntut banyak restoran atau tempat makan untuk menyediakan layanan pemesanan dan pengiriman makanan ke tempat pemesan, di dalam padatnya jalanan ibu kota Jakarta. Katering pada daerah Jakarta merupakan hal penting dalam kehidupan sehari - hari dimana konsumen dapat memesan katering makanan dan minuman tanpa harus keluar rumah. Meningkatnya aktifitas masyarakat yang sama besar dengan kebutuhan pangan menjadi dasar peneliti untuk melakukan penelitian pemodelan dengan *Activity Diagram* untuk perancangan aplikasi pada *platform android* yang dapat memproses pemesanan katering makanan dan minuman dalam jumlah banyak sehingga dapat dilakukan dengan cara lebih praktis serta dapat menghemat waktu dan biaya. Hasil akhir berupa aplikasi pemesanan makanan dan minuman. Aplikasi ini mempunyai beberapa fitur diantaranya, melihat menu katering makanan dan minuman dalam jumlah banyak, aplikasi *E-Commers* katering berguna bagi pengguna untuk memesan katering makanan dan minuman dengan cepat melalui *smartphone android*.

Kata Kunci : Perancangan, Sistem Informasi, *E-Marketpalce Catering*, *Platform Android*.

KATA PENGANTAR

Alhamdulillah Rabbil 'Aalaamiin penulis panjatkan puji syukur ke hadirat Allah Subhanahu Wata'ala atas limpahan rahmat-Nya sehingga penulis dapat menyelesaikan skripsi ini dengan baik. Adapun skripsi berjudul PERANCANGAN DAN IMPLEMENTASI SISTEM INFORMASI *E-MARKETPLACE* UNTUK KATERING, yang penulisambil dapat selesai dengan baik sesuai waktu yang diharapkan.

Untuk menyelesaikan skripsi ini penulis telah banyak mendapatkan bantuan dari berbagai pihak, baik langsung maupun tidak langsung serta bantuan moril maupun materil. Melalui kata pengantar ini dan dengan kerendahan serta ketulusan hati, penulis ingin menyampaikan ucapan terima kasih yang sedalam-dalamnya kepada :

- 1) Ibu Prof. Dr. drg. Tri Budi Wahyuni Rahardjo, MS selaku Rektor Universitas Respati Indonesia.
- 2) Bapak Drs. Djanuri, MM selaku Dekan Fakultas Teknologi Informasi.
- 3) Bapak Andi Susilo S.Kom, M.T.I., selaku Ketua Program Studi Teknik Informatika.
- 4) Bapak Tony Sugiarto, SE, M.Kom selaku dosen pembimbing yang telah bersedia meluangkan waktunya untuk memberikan bimbingan, pengarahan dan koreksi yang bermanfaat dalam penyusunan skripsi ini.
- 5) Jenih, S.Kom, M.Kom selaku dosen Fakultas Teknologi Informasi yang telah bersedia meluangkan waktunya untuk memberikan bimbingan, pengarahan dan koreksi yang bermanfaat dalam penyusunan skripsi ini.

- 6) Seluruh dosen Fakultas Teknologi Informasi yang telah memberikan ilmunya selama proses penulisan skripsi dan semasa kuliah.
- 7) Staff/karyawan/dosen di lingkungan Universitas Respati Indonesia.
- 8) Orang tua dan seluruh keluarga yang telah memberikan doa, dukungan, perhatian, dan kasih sayang yang tulus.
- 9) Para sahabat dan rekan-rekan seperjuangan mahasiswa-mahasiswi Universitas Respati Indonesia yang selalu saling mendukung, bekerja bersama dan memberikan semangat.
- 10) Tidak lupa para alumni Fakultas Teknologi Informatika Universitas Respati Indonesia yang tidak segan membantu serta membagi ilmu dan pengalamannya dalam penulisan skripsi ini.

Penulis sepenuhnya menyadari bahwa dalam penulisan skripsi ini masih jauh dari sempurna. Untuk itu kritik dan saran yang bersifat membangun sangat penulis harapkan demi kesempurnaan penyusunan skripsi ini di masa yang akan datang. Akhir kata semoga penulisan skripsi ini membawa manfaat bagi para pembaca, khususnya mahasiswa Fakultas Teknologi Informasi URINDO.

Jakarta, 16 Juli 2017

Mohammad Andrean Eka Saputra

DAFTAR ISI

	Halaman
Halaman judul	i
Pernyataan Orisinalitas.....	ii
Persetujuan.....	iii
Pengesahan.....	iv
Persetujuan demo program	v
Formulir bimbingan skripsi	vi
Abstrak.....	viii
Kata Pengantar	ix
Daftar Isi	xi
Daftar Tabel	xv
Daftar Gambar	xvi
BAB 1 PENDAHULUAN.....	1
1.1. Latar Belakang	2
1.2. Identifikasi Masalah	3
1.3. Ruang Lingkup	3
1.4. Tujuan penelitian	3
1.5. Manfaat Penelitian	3
1.6. Sistematika Penulisan	4
BAB 2 KAJIAN LITERATUR.....	6
2.1 Kerangka Teori	6
2.1.1 Kutipan teori Dasar	6
A. Pengertian Aplikasi.....	6
B. Sistem Operasi <i>Android</i>	7
C. Katering.....	12
D. <i>Reservasi</i>	14
E. <i>E-Marketplace</i>	14
F. Layanan Antar	17
G. Data	18
H. Normalisasi	22

I. XAMPP	24
2.1.2 Tools Yang Digunakan.....	24
A. Konsep <i>Unified Modelling Language</i>	24
B. Eclipse Integrated Development Environment (IDE)	31
C. <i>Android Development Tools</i> (ADT).....	32
2.2 Kerangka Pikir	32
2.2.1 Penelitian Yang Relevan.....	34
A. Sistem Informasi Pemesanan Makanan dan Minuman pada <i>Omahe Cafe and Resto</i> Berbasis <i>Client Server</i> dengan <i>Platform</i> <i>Android</i>	35
B. Pembuatan Aplikasi Layanan Pesan Antar Makanan pada Sistem Operasi <i>Android</i>	35
2.2.2 Kerangka studi perbandingan.....	36
BAB 3 METODE PENELITIAN.....	38
3.1 Pendekatan Penelitian.....	38
3.1.1 <i>Structured Design</i> model <i>Waterfall</i>	39
3.2 Instrumen Penelitian	42
3.2.1 Studi Kelayakan	42
A. Kelayakan teknis	43
B. Kelayakan ekonomi	43
3.2.2 Metodologi Pengumpulan Data.....	43
A. Wawancara	44
B. Observasi	44
C. Studi Pustaka	44
3.3 Tempat dan Waktu Penelitian.....	44
BAB 4 ANALISIS STRUKTUR PROGRAM APLIKASI.....	45
4.1 Proses Bisnis Program Aplikasi.....	45
4.2 <i>Activity Diagram</i>	46
4.2.1 <i>Activity Diagram Login</i>	46

4.2.2	<i>Activity Diagram Anggota</i>	47
4.2.3	<i>Activity Diagram Search</i>	49
4.2.4	<i>Activity Diagram Catering</i>	50
4.2.5	<i>Activity Diagram About</i>	51
4.2.6	<i>Activity Diagram Pengelola</i>	52
4.2.7	<i>Activity Diagram Lihat Anggota</i>	53
4.2.8	<i>Activity Diagram Produk</i>	54
4.2.9	<i>Activity Order</i>	55
4.2.10	<i>Activity Diagram Profile</i>	56
4.3	Indikasi Aktor	57
4.3.1	<i>Use Case Description</i>	57
4.3.2	<i>Use case Diagram</i>	61
4.3.3	<i>Class Diagram</i>	62
4.3.4	<i>Sequance Diagram</i>	63
4.4	Tujuan Perancangan Program Aplikasi	65
BAB 5 PERANCANGAN PROGRAM APLIKASI		66
5.1	Spesifikasi Bentuk Dokumen Masukan	66
5.2	Spesifikasi Bentuk Dokumen Keluaran	67
5.3	Normalisasi File	69
5.4	Spesifikasi File	74
5.5	Spesifikasi Program	76
5.6	Rancangan Tampilan Layar	77
5.6.1	Rancangan Tampilan Anggota	77
5.6.2	Rancangan Tampilan Pengelola	82
BAB 6 IMPLEMENTASI PROGRAM APLIKASI		87
6.1	Spesifikasi <i>Software</i> dan <i>Hardware</i>	87
6.2	Rancangan Fisik <i>User Interface</i>	88
6.2.1	Tampilan Halaman Default	88
6.2.2	Tampilan Halaman Registrasi Anggota	89
6.2.3	Tampilan <i>Login User</i>	90
6.2.4	Tampilan Daftar Produk	91

6.2.5 Tampilan Pesan Produk	92
6.2.6 Tampilan Detail <i>Order</i>	93
6.2.7 Tampilan Profil Anggota	94
6.2.8 Tampilan <i>Login Admin</i>	95
6.2.9 Tampilan Halaman Daftar <i>Catering</i>	96
6.2.10 Tampilan Daftar Produk	97
6.2.11 Tampilan Halaman Tambah Data Produk	98
6.2.12 Tampilan Halaman Pengguna	99
6.2.13 Tampilan Halaman <i>Order Detail</i>	100
6.3 Peralihan Program Aplikasi	100
BAB 7 PENUTUP	102
7.1 Simpulan.....	102
7.2 Saran	102
DAFTAR PUSTAKA	103
RIWAYAT HIDUP	106

DAFTAR TABEL

Tabel	Nama	Halaman
2.1	Daftar versi <i>android</i>	10
2.2	Studi pembandingan.....	36
4.1	<i>Use case description login</i>	57
4.2	<i>Use case description search</i>	57
4.3	<i>Use case description</i> Daftar <i>E-Marketpalce Catering</i>	58
4.4	<i>Use case description</i> Profil	58
4.5	<i>Use case description</i> order Anggota.....	59
4.6	<i>Use case description</i> About.....	59
4.7	<i>Use case description</i> Tagihan	59
4.8	<i>Use case description</i> order Pengelola	60
4.9	<i>Use case description</i> produk.....	60
4.10	<i>Use case description</i> order profil Pengelola	60
5.1	Spesifikasi file / tabel Pengelola	74
5.2	Spesifikasi file / tabel Anggota	74
5.3	Spesifikasi file / tabel Produk	75
5.4	Spesifikasi file / tabel Order	75
5.5	Spesifikasi file / tabel Order Detail	76

DAFTAR GAMBAR

Gambar	Nama	Halaman
2.1	Model bisnis <i>E-Marketplace</i>	15
2.2	<i>The temple framework</i>	17
2.3	Bagan Kerangka Berpikir	34
3.1	Bagan Model <i>Waterfall</i>	40
4.1	<i>Activity Diagram Login</i>	47
4.2	<i>Activity Diagram Anggota</i>	48
4.3	<i>Activity Diagram Search</i>	49
4.4	<i>Activity Diagram Catering</i>	50
4.5	<i>Activity Diagram About</i>	51
4.6	<i>Activity Diagram Pengelola</i>	52
4.7	<i>Activity Diagram Anggota</i>	53
4.8	<i>Activity Diagram Produk</i>	54
4.9	<i>Activity Diagram Order</i>	55
4.10	<i>Activity Diagram Profile</i>	56
4.11	<i>Use case Diagram E-Marketplace</i>	61
4.12	<i>Class Diagram E-Marketplace Catering</i>	62
4.13	<i>Sequance Diagram Login</i>	63
4.14	<i>Sequence Diagram daftar E-Marketplace Catering</i>	64
4.15	<i>Sequance Diagram Order</i>	64

4.16	<i>Squence Diagram Input Produk</i>	65
5.1	Form Isi <i>Order</i>	66
5.2	Form <i>Input Produk</i>	67
5.3	Bentuk Dokumen Keluaran Tagihan	68
5.4	Bentuk Dokumen Keluaran Produk	68
5.5	Bentuk Tidak Normal	70
5.6	Bentuk Normal Pertama (1NF)	71
5.7	Bentuk Normal Kedua (2NF)	72
5.8	Bentuk Normal Ketiga (3NF)	73
5.9	Rancangan Tampilan <i>Login Anggota</i>	78
5.10	Rancangan Tampilan Menu Utama Anggota	78
5.11	Rancangan Tampilan <i>Search</i>	79
5.12	Rancangan Halaman Daftar <i>E-Marketplace Catering</i>	80
5.13	Rancangan Halaman Daftar Produk	81
5.14	Rancangan Tampilan Halaman Order Anggota	81
5.15	Rancangan Tampilan <i>About</i>	82
5.16	Rancangan Tampilan <i>Login Pengelola</i>	83
5.17	Rancangan Tampilan Halaman Utama Pengelola	83
5.18	Rancangan Tampilan Halaman Profile Pengelola	84
5.19	Rancangan Tampilan Halaman Anggota	84
5.20	Rancangan Tampilan Halaman Produk	85
5.21	Rancangan Tampilan Halaman Order Pengelola	85
5.22	Rancangan Tampilan Halaman Arsip	86

6.1	Tampilan Halaman <i>Default</i>	88
6.2	Tampilan Halaman <i>Registrasi</i>	89
6.3	Tampilan Halaman <i>Login User</i>	90
6.4	Tampilan Halaman Daftar Produk	91
6.5	Tampilan Halaman <i>Order</i>	92
6.6	Tampilan Halaman <i>Detail Order</i>	93
6.7	Tampilan Halaman <i>Profil</i>	94
6.8	Tampilan Halaman <i>Login Admin</i>	95
6.9	Tampilan Halaman Daftar <i>Catering</i>	96
6.10	Tampilan Halaman Daftar Produk	97
6.11	Tampilan Halaman Tambah Produk	98
6.12	Tampilan Halaman Pengguna	99
6.13	Tampilan Halaman <i>Detail Order</i>	100

BAB 1

PENDAHULUAN

Pada bab 1 ini akan dibahas mengenai latar belakang, identifikasi masalah, Ruang lingkup, tujuan dan manfaat, serta sistematika penulisan. Berikut ini penjelasan dari masing-masing sub bab pada pendahuluan.

1.1 Latar Belakang.

Kebutuhan dan permintaan akan jasa penyedia makanan terus meningkat disertai dengan bertambahnya minat masyarakat dan pengusaha untuk ikut berusaha dalam industri kuliner, antara lain dengan membuka usaha katering. Penggunaan katering dalam berbagai acara, banyak digunakan sehingga semakin banyaknya usaha - usaha katering. Dimana hampir semua sektor dalam dunia usaha memanfaatkan kemajuan teknologi untuk mempermudah pekerjaannya. Salah satunya adalah dalam penjualan dan pemasaran produk berupa *e-market*.

E-marketplace sudah banyak bermunculan, walaupun kebanyakan bentuk yang dianut *E-marketplace* di Indonesia pada saat ini adalah berbentuk iklan. Dimana interaktif langsung terjadi antara pembeli dan penjual, sedangkan penyedia *e-marketplace* hanya berfungsi menyediakan tempat saja. Penyedia jasa *e-marketplace* tidak lagi dilibatkan dalam penjualan yang dilakukan oleh pembeli dan penjual. Bentuk *e-marketplace* lainnya adalah lelang, dimana pada bentuk ini *e-marketplace* berfungsi sebagai lelang. Penjual akan memajang barang yang ditawarkan beserta harga awal, lalu pembeli akan mengajukan penawaran. Setelah jangka waktu tertentu, penjual akan memilih tawaran

terbaik, dan pembeli yang mengajukan penawaran tersebut berhak mendapatkan barang yang dijual oleh penjual. Saat ini belum adanya situs *E-marketplace* yang mengkhususkan pada layanan catering. sehingga produk mereka masih banyak yang belum di kenal oleh masyarakat luas. Karena sarana promosi hanya bersifat konvensional yaitu melalui sarana brosur dan mulut – ke mulut.

Berdasarkan permasalahan diatas jasa katering mempermudah dalam proses pemesanan makanan. Dengan adanya aplikasi pemesanan proses operasional katering dapat menjadi efisien. Untuk itu penulis tertarik untuk membuat skripsi yang berjudul, “Perancangan dan Implementasi Sistem Informasi *E-Marketplace* Untuk Katering.

1.2 Identifikasi Masalah.

Berdasarkan latar belakang masalah yang telah diuraikan di atas, maka penulis mengidentifikasi masalah yang ada yaitu:

1. Pengusaha katering mengalami kesulitan dalam mempromosikan dan mengenalkan produk mereka.
2. Pengusaha katering masih memakai cara manual dalam menerima pemesanan.
3. Konsumen kesulitan mencari jasa katering yang dekat dengan lokasi.
4. Konsumen sulit membandingkan antara jasa katering satu dengan yang lain.

1.3 Ruang Lingkup.

Agar penulisan skripsi ini lebih terarah maka penulis membatasi ruang lingkup pembahasan yaitu pada:

- a. Pemrosesan pemesanan produk pada katering.
- b. Adanya laporan *detail order* pada aplikasi.
- c. Terdapatnya dua *user account*, *account Admin* dan *account User*.

1.4 Tujuan Penelitian.

Tujuan dari penelitian ini adalah :

- d. Membantu para pengusaha katering untuk mempromosikan dan menjual produk mereka secara lebih efektif dan efisien kepada konsumen
- e. Memberikan kemudahan kepada konsumen untuk memilih dan membandingkan jasa katering yang sesuai dengan harapan serta terdekat dari lokasi.

1.5 Manfaat Penelitian.

Adapun manfaat dari penulisan skripsi ini adalah :

1. Bagi penulis
 - f. Dapat mengembangkan dan menerapkan ilmu yang diperoleh pada bangku kuliah dalam kehidupan yang nyata.
 - g. Menambah pengetahuan penulis tentang bagaimana merancang sebuah aplikasi marketplace khususnya masalah katering.
2. Bagi pengguna aplikasi
 - a. Sebagai salah satu media alternatif dalam melakukan pemesanan katering makanan dan minuman secara online.
 - b. Dapat mempercepat proses pemesanan katering agar lebih efisien.

3. Bagi Pengusaha Katering
 - a. Dapat menjangkau target ke pasaran yang lebih luas.
 - b. Proses pengelolaan pemesanan akan lebih efisien.
4. Bagi Universitas
 - a. Mengetahui kemampuan mahasiswa dalam menguasai materi kuliah yang telah diperoleh di bangku kuliah.
 - b. Mengetahui kemampuan mahasiswa dalam menerapkan ilmunya sebagai bahan evaluasi kampus.
 - c. Sebagai indikator kesiapan mahasiswa dalam menghadapi dunia kerja yang sebenarnya.

1.6 Sistematika Penulisan.

Adapun sistematika penulisan dalam skripsi ini mencakup hal-hal sebagai berikut :

BAB 1 PENDAHULUAN

Pada bab ini akan diuraikan latar belakang, rumusan masalah, batasan masalah, tujuan dan manfaat, serta sistematika penulisan.

BAB 2 KAJIAN LITERATUR

Pada bab ini akan dijelaskan mengenai kerangka teori dan kerangka pikir yang menjadi landasan dan mendasari penelitian dalam penulisan skripsi ini.

BAB 3 METODE PENELITIAN

Pada bab ini akan dipaparkan mengenai penjelasan Metode penelitian menggunakan pendekatan penelitian, instrumen penelitian dan tempat dan waktu penelitian.

BAB 4 ANALISIS STRUKTUR PROGRAM APLIKASI

Pada bab ini akan dipaparkan tentang bagaimana berjalannya proses bisnis, pemodelan menggunakan *Unified Modeling Language* (UML) dan tujuan perancangan program aplikasi.

BAB 5 PERANCANGAN PROGRAM APLIKASI

Pada bab ini akan dipaparkan mengenai penjelasan desain dan implementasi menyajikan tahap pengembangan program yang akan dijelaskan tiap langkahnya tampilan dari setiap program.

BAB 6 IMPLEMENTASI PROGRAM APLIKASI

Pada bab ini akan dipaparkan mengenai hasil implementasi aplikasi, serta penjelasan tentang perangkat lunak dan perangkat keras yang digunakan.

BAB 7 PENUTUP

Pada bab ini akan dipaparkan kesimpulan dan saran dari aplikasi *marketplace* catering yang telah dibuat untuk pengembangan ke arah yang lebih baik lagi.

BAB 2

KAJIAN LITERATUR

Pada bab ini akan dibahas mengenai kerangka teori dan kerangka pikir yang digunakan untuk kajian teori yang mendukung penulisan. Kerangka teori meliputi kutipan teori dasar yang relevan dengan judul skripsi dan *tools* yang digunakan. Kerangka pikir berisi tentang dasar penelitian yang telah dilakukan maupun penelitian lain sebagai pembandingan.

2.1 Kerangka Teori

Kerangka teori adalah kajian dasar tentang teori yang berhubungan dengan judul dari penulisan. Dalam penulisan kerangka teori, hal yang akan dibahas adalah kutipan teori dasar dan penjelasan *tools* atau *software* yang digunakan dalam perancangan aplikasi.

2.1.1 Kutipan Teori Dasar

Kutipan teori dasar pada penulisan ini mencakup tentang pengertian aplikasi, pengertian sistem operasi *Android*, Katering, Reservasi, Layanan Antar, dan *online*.

A. Pengertian Aplikasi

Menurut Harip (2005), “yang dimaksud dengan pengertian ‘aplikasi’ adalah suatu kelompok file (*form, class, report*) yang bertujuan untuk melakukan aktifitas tertentu yang saling terkait, misalnya aplikasi *fixed asset*, aplikasi *payroll*.” Menurut Hengky (2005), Aplikasi adalah satu unit perangkat lunak yang dibuat untuk melayani kebutuhan akan beberapa aktivitas seperti sistem perniagaan, *game*, pelayanan masyarakat, periklanan, atau semua proses yang hampir dilakukan manusia.

Sedangkan menurut Ali Zaki (2007), Program aplikasi adalah komponen yang berguna melakukan pengolahan data maupun kegiatan-kegiatan seperti pembuatan dokumen atau pengolahan data. Program aplikasi adalah bagian *Personal Computer* yang berinteraksi langsung

dengan *user*. Program aplikasi berjalan diatas sistem operasi, sehingga, agar program aplikasi bisa diaktifkan, anda perlu melakukan instalasi sistem operasi terlebih dahulu.

B. Sistem Operasi *Android*

Sistem operasi *Android* merupakan sistem operasi yang dibuat dengan *Kernel* dari Linux dan didesain khusus untuk perangkat bergerak (*mobile*) seperti *smartphone* dan *tablet*. Sistem operasi ini mendukung sebuah perangkat *mobile* untuk dapat melakukan sesuatu layaknya sebuah telepon pintar. Artinya sistem operasi *Android* hampir secara penuh memfasilitasi seluruh fitur dari sebuah perangkat *mobile* untuk dapat mendukung kinerja manusia dalam berkomunikasi.

Sebuah perangkat yang termasuk kategori *smartphone* secara umum telah dapat melakukan *Browsing*, *real-time video call*, mendukung jaringan 3G – 4G, mendukung Layanan Berbasis Lokasi misalnya *Global Positioning System*, dan mampu *streaming video*. *Android* mendukung lebih dari itu. *Android* mempunyai layanan pendukung yang langsung difasilitasi oleh raksasa internet *Google Inc*. Ini karena *Android* merupakan sistem operasi yang termasuk dalam salah satu proyek *Google Inc*. *Android* mendukung layanan *e-mail* yang terintegrasi dengan *Google Mail*. Layanan *e-mail* ini bersifat multifungsi. Dengan satu alamat *Google Mail* yang telah tersinkronisasi pada perangkat *Android* pengguna, pengguna tersebut dapat melakukan akses secara bebas pada layanan *google* lain seperti *Youtube*, *Google Hangouts*, *Google Maps*, *Google Mail*, *Google Translate*, sampai *Google Playstore* . Dengan kata lain alamat *e-mail* tersebut menjadi identitas resmi bagi pengguna *Android*.

Layaknya komputer mini. *Android* juga mendukung aplikasi pengolahan dokumen seperti presentasi, pengolah kata, pengolah gambar, pengolah angka dan lainnya dapat berfungsi dengan baik. Tidak ketinggalan aplikasi untuk hiburan seperti *game*. *Game* dengan tingkat resolusi dan animasi 3D dapat dijalankan dengan baik pada perangkat *Android* yang didukung dengan spesifikasi *hardware* untuk grafis tingkat

tinggi dari bagian *Random Access Memory*, *processor*, resolusi layar dan lainnya. Layanan *Google Playstore* membantu dalam menemukan aplikasi yang cocok bagi kebutuhan sehari-hari penggunanya.

a. Sejarah Sistem Operasi *Android*

Android sebagai sebuah sistem operasi yang dikembangkan untuk perangkat *mobile* mempunyai basis *kernel* yang berasal dari Linux. Cakupan dari sistem operasi ini meliputi sistem operasi, *middleware*, dan aplikasi. *Android* menyediakan *platform* terbuka bagi para pengembang aplikasi untuk menciptakan aplikasi mereka.

Sejak resmi dirilis pada tahun 2007, *Android* telah banyak mengalami perubahan dari segi versi dan kemampuan. Awalnya sistem operasi *Android* dikembangkan oleh *Android Inc* yang merupakan pendatang baru dalam pembuatan peranti lunak untuk *smartphone*, kemudian dibeli oleh *Google Inc*. Lalu untuk mengembangkan *Android*, dibentuklah *Open Handset Alliance*, konsorsium dari 34 perusahaan peranti keras, peranti lunak, dan telekomunikasi, yang termasuk di dalamnya adalah *Google Inc*, *HTC*, *Intel*, *Motorola*, *Qualcomm*, *T-Mobile*, *NVidia*, dan lain sebagainya.

Pada saat perilis perdana *Android*, 5 November 2007, *Android* bersama *Open Handset Alliance* menyatakan mendukung pengembangan *open source* pada perangkat *mobile*. Di lain pihak, *Google* merilis kode-kode *Android* dibawah lisensi *Apache*, sebuah lisensi perangkat lunak dan *open platform* perangkat seluler.

Di dunia ini terdapat dua jenis distributor sistem operasi *Android*. Pertama yang mendapatkan dukungan penuh dari *Google* atau *Google Mail Service* (GMS) dan kedua adalah yang benar-benar bebas distribusinya tanpa dukungan langsung *Google* atau yang lebih dikenal sebagai *Open Handset Distribution* disingkat OHD.

Sekitar September 2007, Google mengenalkan *Nexus One*, salah satu jenis *smartphone* yang menggunakan *Android* sebagai sistem operasinya. Telepon seluler ini diproduksi oleh *HTC Corporation* dan tersedia di pasaran pada 5 Januari 2010. Pada 9 Desember 2008, diumumkan anggota baru yang bergabung dalam program kerja *AndroidARM Holdings*, *Atheros Communications*, diproduksi oleh *Asustek Computer Inc*, *Garmin Ltd*, *Softbank*, *Sony Ericsson*, *Toshiba Corp*, dan *Vodafone Group Plc*. Seiring pembentukan *Open Handset Alliance*, *Open Handset Alliance* mengumumkan produk perdana mereka, *Android*, perangkat *mobile* yang merupakan modifikasi kernel Linux 2.6. Sejak *Android* dirilis telah dilakukan berbagai pembaruan berupa perbaikan *bug* dan penambahan fitur baru.

Pada masa saat ini, sebagian besar *vendor-vendor smartphone* sudah memproduksi *smartphone* berbasis *Android*. *Vendor-vendor* itu antara lain *HTC*, *Motorola*, *Samsung*, *LG*, *HKC*, *Huawei*, *Archos*, *Webstation Camangi*, *Dell*, *Nexus*, *SciPhone*, *WayteQ*, *Sony Ericsson*, *Acer*, *Phillips*, *T-Mobile*, *Nexian*, *IMO*, *Asus*, dan masih banyak lagi *vendor smartphone* di dunia yang memproduksi *Android*. Hal ini, karena *Android* itu adalah sistem operasi yang *open source* sehingga bebas didistribusikan dan dipakai oleh *vendor* manapun.

Tidak hanya menjadi sistem operasi di *smartphone*, saat ini *Android* menjadi pesaing utama dari *Apple* pada sistem operasi *Tablet*. Pesatnya pertumbuhan *Android* selain faktor yang disebutkan diatas adalah karena *Android* itu sendiri adalah *platform* yang cukup lengkap baik itu sistem operasinya, aplikasi dan *Tool* pengembangan, *application market* serta dukungan yang sangat tinggi dari komunitas *open source* di dunia, sehingga *Android* terus

berkembang pesat baik dari segi teknologi maupun dari segi jumlah *device* yang ada di dunia.

Telepon seluler pertama yang memakai sistem operasi *Android* adalah HTC Dream, yang dirilis pada 22 Oktober 2008. Pada penghujung tahun 2010 diperkirakan hampir semua *vendor* seluler di dunia menggunakan *Android* sebagai sistem operasi. *Android* sejak pertama dirilis telah merilis beberapa versi sampai sekarang. Penamaan versinya terbilang cukup unik, yaitu menggunakan nama-nama camilan yang manis dan bergula. Adapun versi-versi yang pernah dirilis adalah sebagai berikut :

Tabel 2.1 Daftar versi *android*

Versi	Tanggal Rilis	Fitur Tambahan Utama
<i>Android</i> 1.0	23 September 2008	Perangkat komersial pertama yang menggunakan versi <i>Android</i> pertama ini adalah HTC Dream. Pada versi ini google telah memberikan fitur-fitur seperti <i>browser</i> , Gmail, <i>Google Maps</i> , Google Calendar, Google Contact, Gtalk, Youtube, juga layanan sinkronisasi google (<i>google Sync</i>) pada ponsel.
<i>Android</i> 1.1	9 Februari 2009	Penambahan pemberitahuan email, pencarian suara
<i>Android</i> 1.5 (Cupcake)	30 April 2009	Dukungan <i>virtual keyboard</i> dengan prediksi teks dan kamus, Support bluetooth A2DP & AVRCP, kemampuan <i>record&watch</i> Video, mengunggah video ke youtube dan gambar ke picasa.
<i>Android</i> 1.6 (Donut)	15 September 2009	Fitur pencarian yang diperbaharui, Mesin sintesis pengucapan multibahasa, <i>Gesture Framework</i> , dukungan CDMA/EVDO, 802.1x, VPN.
<i>Android</i> 2.0 - 2.1 (Eclair)	26 Oktober 2009 –	Bluetooth 2.1, <i>Live wallpaper</i> , dukungan HTML5, penambahan

	12 Januari 2010	fitur kamera termasuk <i>flash</i> , <i>zoom</i> , <i>dll</i> . Sinkronisasi akun, peningkatan <i>Google Mapss</i> , <i>Microsoft Exchange Support</i> .
<i>Android 2.2 - 2.2.1</i> (Froyo: Frozen Yoghurt)	20 Mei 2010 – 21 November 2011	Peningkatan kecepatan, penyimpanan, dan pengoptimalan kinerja. Integrasi mesin Javascript V8 Chrome pada aplikasi Browser. Dukungan A2DM, Dukungan Adobe Flash.
<i>Android 2.3 – 2.3.7</i> (Gingerbread)	6 Desember 2010 – 21 September 2011	Update UI, Peningkatan fitur copy paste, Dukungan obrolan video dan suara di Google Talk.
<i>Android 3.0 – 3.2</i> (Honeycomb)	22 Februari 2011 – Februari 2012	Pembaruan <i>Android</i> pertama yang ditujukan hanya untuk <i>Tablet</i> , perangkat pertama yang menggunakan versi ini adalah Motorola Xoom. Karena dirancang khusus untuk <i>Tablet</i> , maka antarmuka nya juga berbeda dengan versi HH (<i>HandHeld</i>). Mendukung <i>multi-core processor</i> .
<i>Android 4.0 – 4.0.4</i> (Ice Cream Sandwich)	19 Oktober 2011 – 29 Maret 2012	Dibangun dengan basis linux kernel 3.0.1, fitur utama yang ditambahkan antara lain : Pemisahan <i>widget</i> di tab baru, diletakkan bersebelahan dengan <i>Application Tab</i> . Membuat <i>Icon Folder</i> lebih mudah dengan <i>drag-and-drop</i> . <i>Launcher</i> yang bisa di kustomisasi. <i>Screenshot</i> layar dengan menekan tombol <i>volume</i> naik/turun dan tombol <i>on/off</i> . Modul kernel <i>Android VPN Framework</i> (AVF) dan TUN. Rotasi layar yang lebih halus, dan perbaikan-perbaikan bug sebelumnya.
<i>Android 4.1 – 4.3</i> (Jellybean)	9 Juli 2012 – 24 Juli 2013	Perangkat pertama yang menggunakan OS ini adalah Nexus 7. Berdasarkan kernel linux 3.0.31, Jelly Bean dibangun untuk peningkatan kinerja dan fungsi antar muka pengguna (<i>user-</i>

		<i>interface</i>). Peningkatan <i>frame rate</i> hingga 60 fps untuk menciptakan UI yang lebih halus. <i>Keyboard</i> yang bisa dimodifikasi, perluasan notifikasi, <i>browser</i> bawaan diganti menjadi <i>Chrome</i> .
Android 4.4 (Kit Kat)	31 Oktober 2013	Optimasi kinerja dengan perangkat spesifikasi lebih rendah. Pembaruan antar muka, <i>NFC Host Card Emulation</i> sebagai emulator kartu pintar NFC.
Android 5.0 (Lollipop)	3 November 2014	Desain antarmuka (tampilan) yang dinamakan " <i>Material Design</i> ", 64-bit ART <i>compiler</i> , <i>project volta</i> , yang berguna untuk meningkatkan daya hidup baterai 30% lebih tahan lama, ' <i>factory reset protection</i> '. Fitur ini berguna ketika <i>smartphone</i> hilang, ia tidak bisa direset ulang tanpa memasukkan <i>id google</i> dan kata sandi (<i>password</i>)
Android 6.0 (Marshmallow)	5 Oktober 2015	Dukungan USB Type-C. Dukungan otentifikasi sidik jari. Mode "deep sleep" untuk menghemat daya baterai. <i>Permission dashboard</i> . <i>Android Pay</i> . Peningkatan fitur <i>Google now</i>

Sumber : Wikipedia Indonesia.

C. Katering

Davis dan Store dalam Kardigantara (2006:4), jasa boga (katering) termasuk dalam industri *Commercial Katering* yaitu maksud dan tujuan dari perusahaannya adalah untuk mendapatkan profit melalui jasa layanan katering yang bertujuan memenuhi dan

memuaskan kebutuhan konsumen melalui produk (jasa) yang disediakan. Wikipedia Indonesia dalam Ensiklopedia bebas berbahasa Indonesia : Jasa boga atau yang lebih dikenal dengan catering adalah istilah umum untuk wirausaha yang melayani pemesanan berbagai macam masakan (makanan dan minuman) baik untuk pesta maupun untuk suatu instansi.

a. Tujuan Catering

Beberapa tujuan yang dapat dicapai melalui penyediaan layanan catering daring :

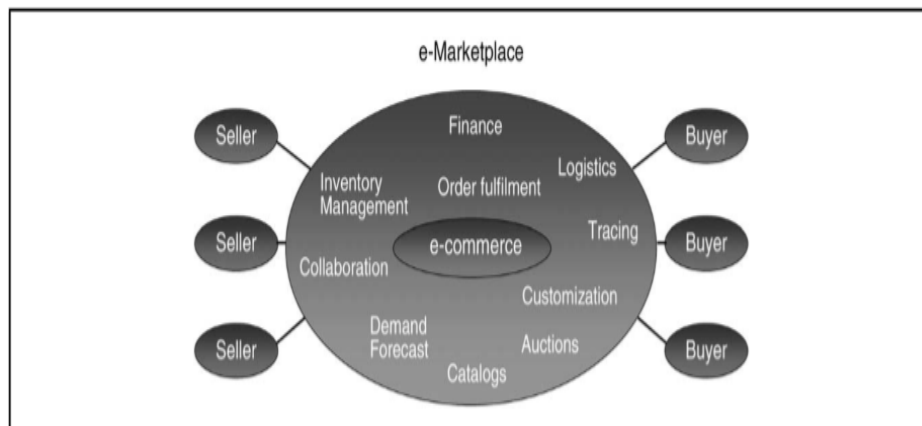
1. memberikan kesempatan kepada konsumen untuk belajar memilih makanan yang baik atau sehat.
2. memberikan bantuan dalam mengajarkan ilmu gizi secara nyata.
3. menganjurkan kebersihan dan kesehatan.
4. menekankan penggunaan tata krama yang benar dan sesuai dengan yang berlaku di masyarakat.
5. memberikan gambaran tentang manajemen yang praktis dan baik.
6. menunjukkan adanya koordinasi antara bidang pertanian dengan bidang industri.
7. menghindari terbelinya makanan yang tidak dapat dipertanggungjawabkan kebersihannya dan kesehatannya.

D. Reservasi

- a. Pemesanan dalam bahasa Inggris adalah *reservation* yang berasal dari kata *to reserve* yaitu menyediakan atau mempersiapkan tempat sebelumnya. Sedangkan *reservation* yaitu pemesanan suatu fasilitas.
- b. 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 kesepakatan mengenai produk dapat terwujud. (Suartana, 1987).

E. E-Marketplace

E-marketplace merupakan bagian dari *e-commerce*. Menurut Brunn, Jensen, & Skovgaard (2002), *e-marketplace* adalah wadah komunitas bisnis interaktif secara elektronik yang menyediakan pasar dimana perusahaan dapat ambil andil dalam B2B *e-commerce* dan atau kegiatan *e-business* lain. Pada gambar 2.1, *e-marketplace* dapat dikatakan sebagai gelombang kedua pada *e-commerce* dan memperluas kombinasi dari bisnis konsumen (B2B,C2B dan C2C) ke dalam B2B. Inti penawaran dari *e-marketplace* adalah mempertemukan pembeli dan penjual sesuai dengan kebutuhan dan menawarkan efisiensi dalam bertransaksi.



Gambar 2.1 Model bisnis *E-Marketplace*.

Sumber : Brunn, Jensen, & Skovgaard (2002)

Menurut Brunn, Jensen, & Skovgaard (2002) terdapat dua jenis *e-marketplaces*:

a. *E-marketplaces Horizontal*

E-marketplaces horizontal dikategorikan berdasarkan fungsi atau produk umum yang ditawarkan perusahaan. Dapat diartikan pasar yang digunakan untuk industri umum. Seperti pasar penjualan *smartphone*, *pc*, baju. dikeluarkan lebih rendah.

b. *E-marketplaces vertical*

E-marketplaces vertical dapat diartikan pasar yang digunakan untuk industri yang memenuhi kebutuhan khusus pada masing - masing industri. Seperti pasar penjualan beton, baja.

a. **Strategi *E-Marketplace***

Dalam pengaplikasian *e-marketplace* dibutuhkan strategi untuk mengoptimalkan *e-marketplace*. Gambar 2.2 menjelaskan tentang *the temple framework e-marketplace*. Menurut Brunn, Jensen, & Skovgaard (2002) terdapat tiga bagian utama yaitu :

1. Pengaturan

Apa saja yang menjadi pondasi suksesnya *e-marketplace* yaitu fokus, pemerintahan, fungsi, teknologi dan kerjasama. Fokus pada bisnis tentu perlu, perusahaan harus memiliki konsep kuat

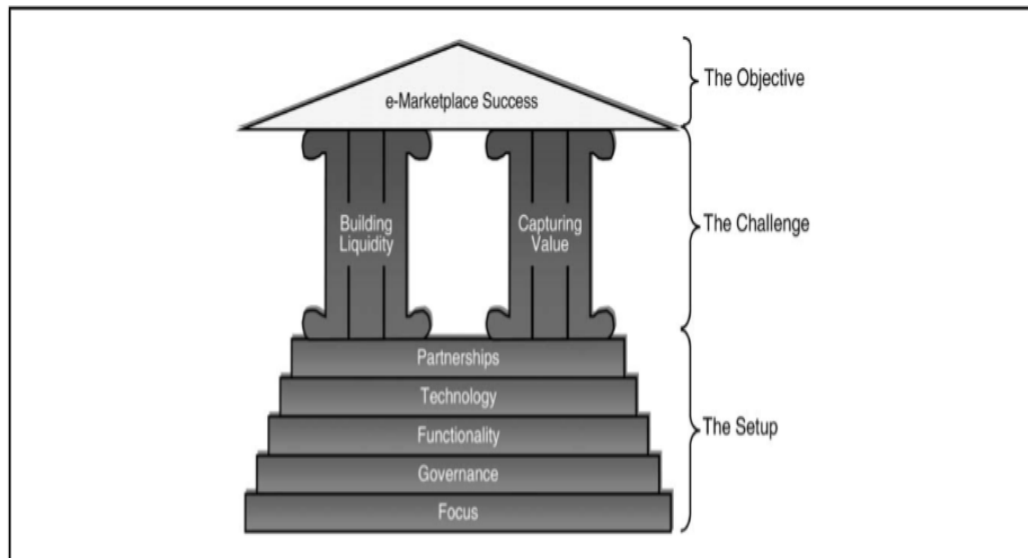
dan fokus akan target yang ingin dicapai. Pemerintahan dalam arti membangun perusahaan memerlukan adaptasi dengan peraturan-peraturan yang dibuat oleh pemerintah setempat. Fungsionalitas diartikan sebagai produk atau jasa yang ditawarkan memiliki fungsi yang tepat dan sesuai kebutuhan pasar. Untuk teknologi, teknologi terus berkembang, perusahaan harus dapat beradaptasi dengan teknologi untuk terus bertahan. Kerjasama adalah salah satu faktor penting untuk membangun *e-marketplace* yang sukses, karena dengan menjalin kerjasama yang baik dengan berbagai pihak apalagi pihak utama perusahaan maka diharapkan proses bisnis perusahaan lebih lancar. Semua pengaturan menjadi pondasi yang harus kuat bagi perusahaan. Meskipun pengaturan ini bersifat dinamis karena disesuaikan dengan lingkungan perusahaan.

2. Tantangan

Perusahaan diharapkan untuk membangun likuiditas dan menangkap nilai sebagai tantangan. Kedua hal tersebut saling berhubungan erat satu sama lain. Untuk itu diperlukan pemikiran yang tepat untuk menjaga kedua hal tersebut agar perusahaan dapat terus bertahan hingga tercapai sukses.

3. Tujuan

Pengaturan dan tantangan harus didiskusikan dengan baik karena sebagian *e-marketplace* masih dalam tahap awal. Perlu dipikirkan isu-isu yang berkaitan dengan *e-marketplace* sehingga dapat ditemukan solusi dan tujuan *e-marketplace* yang sukses dapat tercapai



Gambar 2.2 *The temple framework.*

Sumber : Brunn, Jensen, & Skovgaard (2002)

Penggunaan *e-marketplace* pada perusahaan berdampak positif bagi beberapa aspek, namun *e-marketplace* juga memiliki keterbatasan. dan batasan. Gambar 2.3 mendeskripsikan keuntungan dan batasan dari penggunaan *e-marketplace*.

F. Layanan Antar

Pengertian Layanan antar (*Home delivery service*) menurut beberapa ahli yaitu :

- a. *Home-delivery services where meals are delivered at home of the person who orders the meals.* Sudhir Andrews (2009)
- b. *Where prepared food is brought to the customer's home"* Kaye Chon dan Thomas A. Maier (2010)
- c. *Delivery service relies heavily on telephone orders, with an increasing number of restaurants accepting delivery order via the internet.* Regina S. Barbaran dan Joseph F. Durocher (2010).

Dari beberapa pengertian di atas, maka penulis menyimpulkan bahwa pengertian layanan antar adalah suatu aktivitas dan pemberian jasa dimana *customers* memesan produk yang disediakan produsen dan biasanya menggunakan media komunikasi melalui telepon atau internet lalu produk yang dipesan akan diantarkan sampai ke tempat tujuan *customers*.

G. Data

Istilah data berasal dari bahasa latin "DATUM" yang berarti fakta yang mengandung arti dan dihubungkan dengan kenyataan, simbol, angka, gambar, huruf, kata dan lain-lain. Definisi dari data adalah merupakan bentuk yang masih mentah yang belum dapat bercerita banyak, sehingga perlu diolah lebih lanjut. Data diolah melalui suatu model untuk menghasilkan informasi.

a. Pengolahan Data

Pengolahan data adalah serangkaian operasi atas informasi yang direncanakan guna mencapai tujuan atau hasil yang diinginkan. Tujuan utama dalam pengolahan data dalam sebuah database adalah agar kita dapat memperoleh kembali data yang kita cari dengan mudah dan cepat, selain itu pemanfaatan database memiliki beberapa tujuan. Secara lengkap pemanfaatan database dilakukan untuk memenuhi sejumlah tujuan objektif antara lain :

1. Kecepatan dan Kemudahan (*Speed*)
2. Efisiensi ruang (*Space*)
3. Ketersediaan (*Availability*)
4. kelengkapan (*Completely*)
5. keamanan (*Security*)
6. Kebersamaan Pemakai (*Sharability*)

b. Basis Data (*Database*)

Basis Data (*Database*) adalah sekumpulan informasi bermanfaat yang diorganisasikan ke dalam tata cara yang khusus. Komponen utama Basis Data (*Database*) Antara lain :

1. Perangkat Keras (*Hardware*)
Yaitu Sebuah Komputer yang sudah berbentuk PC.
2. Sistem Operasi (*Software*)
Yaitu suatu bahasa pemrograman untuk melayani perintah – perintah *user*.
3. Data
Yaitu data yang bersifat terpadu dan berbagi.
4. Aplikasi
5. Yaitu pengolahan perangkat lunak atau sistem informasi yang bersifat opsional.
6. Pemakai (*User*)
Yaitu terdiri dari *End User* dan *database administrator*.

Dalam Basis Data Sistem informasi digambarkan dalam model *Entity Relationship* (E-R). Bahasa yang digunakan dalam Basis Data yaitu :

1. DDL (*Data Definition Language*)
Merupakan bahasa definisi data yang digunakan untuk membuat dan *manage* objek *database* seperti *database*, tabel dan *view*.
2. DML (*Data Manipulation Language*)
Merupakan bahasa manipulasi data yang digunakan untuk memanipulasi data pada objek *database* seperti tabel.
3. DCL (*Data Control Language*)
Merupakan bahasa yang digunakan untuk mengendalikan pengaksesan data.

Menurut *Connolly* dan *Begg* (2010:65), *database* adalah sekumpulan data tersebar yang berhubungan secara logis, dan penjelasan dari data ini dirancang untuk memenuhi kebutuhan informasi dari suatu organisasi.

Menurut *Inmon* (2005:493), *database* adalah sekumpulan data yang saling berhubungan yang disimpan, dengan redundansi yang terkontrol dan terbatas berdasarkan skema. Sebuah *database* dapat melayani *single* atau *multiple applications*.

Menurut *Gottschalk* dan *Saether* dalam jurnal (2010:41), *database* adalah sekumpulan data yang terorganisir untuk mendukung banyak aplikasi secara efisien dengan memusatkan data dan mengontrol *data redundant*.

Berdasarkan definisi-definisi yang dijabarkan oleh para ahli di atas, maka dapat disimpulkan *database* adalah sekumpulan data yang saling berhubungan dan terorganisir yang disimpan berdasarkan skema dengan memusatkan data dan mengontrol *data redundant* untuk memenuhi kebutuhan informasi dari suatu organisasi.

1) ***Data Base Management System (DBMS)***

Menurut *Connolly & Begg* (2010:66), *Database Management System* adalah suatu sistem perangkat lunak yang memungkinkan pengguna untuk mendefinisikan, membuat, memelihara, dan mengontrol akses ke *database*.

Manajemen Sistem Basis Data (Database Management System / DBMS) adalah perangkat lunak yang didesain untuk membantu dalam hal pemeliharaan dan utilitas kumpulan data dalam jumlah besar. DBMS dapat menjadi alternatif penggunaan secara khusus untuk aplikasi, semisal penyimpanan data dalam file dan menulis kode aplikasi yang spesifik untuk pengaturannya. Penggunaan DBMS untuk

suatu aplikasi tergantung pada kemampuan dan dukungan DBMS yang beroperasi secara efisien. Sehingga agar bisa menggunakan DBMS dengan baik, perlu diketahui cara kerja dari DBMS tersebut. Pendekatan yang dilakukan untuk menggunakan DBMS secara baik meliputi : implementasi DBMS dan arsitektur secara mendetail untuk dapat memahami desain dari suatu basis data

Berikut adalah komponen *Database Management System* menurut *Connolly & Begg* (2010:68) yang terdiri atas 5 komponen, yaitu :

1. *Hardware*

Database Management System membutuhkan *hardware* untuk menjalankan aplikasi–aplikasinya yang meliputi *Personal Computer*, *mainframe*, dan suatu jaringan komputer.

2. *Software*

Komponen perangkat lunak yang meliputi *software Database Management System* itu sendiri, program aplikasi, sistem operasi, termasuk dan sistem jaringan.

3. *Data*

Data merupakan komponen yang paling penting dalam *Database Management System*, berasal dari sudut pandang dari *end – user*. Data berperan sebagai penghubung antara mesin dan pengguna.

4. *Prosedur*

Prosedur merupakan instruksi dan aturan yang mengatur perancangan dan penggunaan *database*.

5. *People*

Komponen terakhir adalah manusia yang berhubungan langsung dengan sistem. Komponen ini meliputi

Database Administrator, Database Designers, Application Developers, dan end – user.

2) **MySql**

Menurut Antonius Nugraha Widhi Pratama (2010:10) *MySQL* adalah suatu sistem manajemen basis data relasional yang mampu bekerja dengan cepat, kokoh dan mudah digunakan. *MySQL* pertama kali dirintis oleh seorang *programer database* bernama *Michael Widenius*. Selain *MySQL* ada beberapa program *database server* lain yang menggunakan standar *query* berupa *SQL*, antara lain adalah:

- a) *Oracle*
- b) *PostgreSQL*
- c) *MySQL front*
- d) *MySQL*
- e) *SQL Server 97*

MySQL adalah sebuah *database server*, dapat juga berperan sebagai *client* sehingga sering disebut *database client*, yang *open source* dengan kemampuan dapat berjalan baik di *Operating System* manapun, dengan *Platform Windows* maupun *Linux*.

H. Normalisasi

Normalisasi adalah proses mendesain struktur Database dan teknik analisis data yang mengorganisasikan atribut data dengan cara mengelompokkan sehingga terbentuk entitas yang non-redundant, stabil, serta fleksible sehingga menghasilkan sebuah table yang normal.

Normalisasi merupakan teknik analisis data yang mengorganisasikan atribut-atribut data dengan cara mengelompokkan sehingga terbentuk entitas yang non-redundant, stabil, dan fleksible.

Normalisasi dilakukan sebagai uji coba pada suatu relasi secara berkelanjutan untuk menentukan apakah relasi itu sudah baik, yaitu dapat dilakukan proses insert, update, delete, dan modifikasi pada satu atau beberapa atribut tanpa mempengaruhi integritas data dalam relasi tersebut.

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

1) **Normal Form**

Data yang direkam dan dimasukkan secara mentah dalam suatu tabel pada bentuk ini sangat mungkin terjadi inkonsistensi dan anomali data. Pada bentuk ini ada beberapa ciri-ciri yang penting, yang pertama adalah akan terjadi anomali dalam insert, update, dan delete. Hal ini menyebabkan beberapa fungsi DML dalam SQL tidak dapat berjalan dengan baik. Sebagai contoh jika ingin menghapus penerbit maka data judul buku akan ikut terhapus begitu juga jika ingin menghapus peminjam, maka data penerbit dan buku yang harusnya tidak terhapus akan ikut hilang.

2) First Normal Form (1NF)

Bentuk normal yang pertama atau 1NF mensyaratkan beberapa kondisi dalam sebuah database, berikut adalah fungsi dari bentuk normal pertama ini. Menghilangkan duplikasi kolom dari tabel yang sama. Buat tabel terpisah untuk masing-masing kelompok data terkait dan mengidentifikasi setiap baris dengan kolom yang unik (primary key).

3) Second normal form (2NF)

Syarat untuk menerapkan normalisasi bentuk kedua ini adalah data telah dibentuk dalam 1NF, berikut adalah beberapa fungsi normalisasi 2NF. Menghapus beberapa subset data yang ada pada tabel dan menempatkan mereka pada tabel terpisah, Menciptakan hubungan antara tabel baru dan tabel lama dengan menciptakan foreign key. Tidak

ada atribut dalam tabel yang secara fungsional bergantung pada candidate key tabel tersebut.

I. **XAMPP**

Menurut Yogi Wicaksono (2008:7) *XAMPP* adalah sebuah *software* yang berfungsi untuk menjalankan *website* berbasis *Personal Home Page* dan menggunakan pengolah data *MySQL* di komputer lokal". *XAMPP* berperan sebagai *server web* pada komputer anda. *XAMPP* juga dapat disebut sebuah *CPanel server virtual*, yang dapat membantu anda melakukan *preview* sehingga dapat memodifikasi *website* tanpa harus *online* atau terakses dengan *internet*.

2.1.2 **Tools Yang Digunakan**

Dalam sub-bab ini akan dibahas tentang *tools* atau alat bantu yang digunakan untuk merancang aplikasi dalam penulisan ini. *Tools* yang digunakan adalah struktur pemodelan menggunakan *Unified Modelling Language* (UML) dan perangkat lunak perancang aplikasi atau *application development tool*. Perangkat lunak yang digunakan adalah *Eclipse Integrated Development Environment* dengan *plugin* nya yaitu *Android Development Tools*.

A. **Konsep Unified Modelling Language**

Unified Modelling Language disingkat UML adalah sebuah bahasa yang telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak. *Unified Modelling Language* menawarkan sebuah standar untuk merancang mode sebuah sistem. (Verdi Yasin, 2012)

Kita dapat membuat model apapun untuk semua jenis aplikasi *software* dengan UML. Model dari aplikasi tersebut dapat berjalan pada *hardware*, sistem operasi, dan jaringan apapun, serta dapat ditulis dengan bahasa pemrograman apapun. Namun karena dalam konsep dasar UML menggunakan *class* dan *operation*, UML lebih cocok untuk penulisan

software menggunakan bahasa pemrograman berorientasi objek seperti C++, Java, C#, atau VB.NET. walaupun demikian, UML tetap cocok digunakan untuk *modelling* dengan menggunakan bahasa lain seperti VB atau C.

UML mempunyai notasi dan *syntax*/semantik. Notasi pada UML memiliki bentuk dengan arti tertentu yang digunakan untuk menggambarkan diagram pada pemodelan. UML *Syntax* mendefinisikan bagaimana bentuk-bentuk notasi tersebut dapat dikombinasikan. Notasi UML yang ada sekarang merupakan turunan dari 3 notasi yang telah ada sebelumnya yaitu dari Grady Booch *Object-Oriented Design* (OOD), Jim Rumbaugh *Object Modeling Technique* (OMT), dan Ivar Jacobson *Object-Oriented Software Engineering* (OOSE)

a. Sejarah UML

Menurut Yasin, Verdi. *Rekayasa Perangkat Lunak Berorientasi Objek, Pemodelan, Arsitektur dan Perancangan (Modelling, Architecture and Design)*(2012). Bahasa pemodelan berorientasi objek muncul antara sekitar pertengahan tahun 1970-an dan akhir tahun 1980-an yang dikenal dengan bahasa pemrograman berorientasi objek dan aplikasi kompleks yang berkembang, yang dimulai untuk eksperimen dengan pendekatan alternatif untuk analisis dan desain. Sejumlah metode berorientasi objek bertambah dari kurang lebih 10 sampai lebih dari 50 selama periode 1989 dan 1994.

Beberapa *user* pengguna metode ini menemukan permasalahan dalam bahasa pemodelan ini yang dibutuhkan mereka untuk kelengkapan, sehingga timbul yang dinamakan perang metode yang terkemuka, seperti Booch, Jacobson's *Object Oriented Software Engineering* (OOSE) dan Rumbaugh's *Object Modelling Technique* (OMT). Metode penting lainnya seperti Fusion, Shier-Mellor dan Coad-Yourdan. Setiap metode ini merupakan metode

yang lengkap, meskipun setiap metode diakui memiliki kelebihan dan kekurangan. Dalam waktu yang singkat metode *Booch* paling terasa dalam mendesain dan membangun tahapan *project*, *Object Oriented Software Engineering* memberikan dukungan yang baik untuk *use cases* seperti cara untuk menjalankan permintaan, analisis dan desain level tinggi, dan *Object Modelling Technique -2* sangat berguna untuk analisis dan sistem informasi data intensif.

Banyak ide-ide yang kritis dimulai dari pertengahan tahun 1990-an ketika Grady Booch (*Relational Software Corporation*), Ivar Jacobson (*Objectory*) dan James Rumbaugh (*General Electric*) mulai mengadopsi ide-ide dari metode lainnya yang dikumpulkan yang akhirnya diakui sebagai Metode Object Oriented yang mudah di seluruh dunia. Kemudian mereka termotivasi untuk membangun *Unified Modelling Language (UML)*.

Ada 3 tujuan dibangunnya penyatuan metode tersebut yaitu :

- 1) Untuk memodelkan sistem, dari konsep ke bentuk yang cocok dengan menggunakan teknik berorientasi objek.
- 2) Untuk menunjukkan skala persoalan kompleks.
- 3) Untuk membangun bahasa pemodelan yang berguna bagi manusia dan mesin.

Perencanaan bahasa untuk digunakan pada analisis dan desain yang berorientasi objek tidak seperti mendesain bahasa pemrograman. Pertama, kita harus mengetahui masalah seperti dapatkah bahasa mencakup spesifikasi permintaan? Dapatkah bahasa penting untuk pemrograman visual? Kedua, kita harus menemukan keseimbangan antara kompleks dan kesederhanaan. Bahasa yang terlalu sederhana akan terbatas untuk masalah yang luas yang akan dipecahkan. Sedangkan untuk bahasa yang kompleks akan berakibat terlalu mengembang pada sistem yang sederhana.

Unified Modelling Language dimulai secara resmi pada Oktober 1994, ketika Rumbaugh bergabung dengan Booch pada *Relational Software Cooperation*. Proyek ini memfokuskan pada penyatuan metode Booch dan *Object Modelling Technique*. Versi 0.8 merupakan metode penyatuan yang dirilis pada bulan Oktober 1995. Dalam waktu yang sama Jacobson bergabung dengan *Relational Software Cooperation* dan cakupan UML semakin luas sampai di luar perusahaan OOSE. Dokumentasi *Unified Modelling Language* versi 0.9 akhirnya dirilis pada bulan Juni 1996. Dalam waktu tersebut menjadi lebih jelas bahwa beberapa organisasi *software* melihat kalau *Unified Modelling Language* merupakan strategi dari bisnisnya. Kemudian dibangunlah *Unified Modelling Language Consortium* dengan beberapa organisasi yang menyumbangkan sumber dayanya untuk bekerja mengembangkan dan melengkapi *Unified Modelling Language*.

Pada dasarnya sejarah *Unified Modelling Language* cukup panjang. Sampai era tahun 1990 seperti kita ketahui puluhan metodologi pemodelan berorientasi objek telah bermunculan di dunia diantaranya metodologi Booch, metodologi Coad, metodologi OOSE, Metodologi OMT, Metodologi Shlaer-mellor, Metodologi wirfs-brock, dan sebagainya. Pada masa itu terkenal dengan masa perang metodologi dalam pendesainan pemodelan berorientasi objek. Masing-masing metodologi membawa notasi sendiri-sendiri yang mengakibatkan timbul masalah baru jika kita bekerjasama dengan group/perusahaan lain yang menggunakan metodologi yang berlainan.

b. Tujuan UML

Penggunaan UML bertujuan untuk :

- 1) Memodelkan suatu sistem (bukan hanya perangkat lunak) yang menggunakan konsep berorientasi objek.

- 2) Menciptakan suatu bahasa pemodelan yang dapat digunakan baik oleh manusia maupun mesin.

c. Diagram-diagram Dalam *Unified Modelling Language*

Umumnya sebuah sistem mempunyai sejumlah *stakeholder* (orang yang mempunyai ketertarikan pada suatu sistem namun dari sudut pandang yang berbeda). *Unified Modelling Language* memiliki sejumlah elemen grafis yang bisa dikombinasikan menjadi diagram. Karena ini merupakan bahasa, *Unified Modelling Language* mempunyai sejumlah aturan untuk menggabungkan elemen-elemen tersebut. Meskipun *Unified Modelling Language* sudah cukup banyak menyediakan diagram yang bisa membantu mendefinisikan sebuah aplikasi, tidak berarti bahwa semua diagram tersebut akan bisa menjawab persoalan yang ada. Dalam banyak kasus, diagram selain *Unified Modelling Language* sangat banyak membantu. Oleh karena itu jangan ragu-ragu untuk menggunakan diagram selain *Unified Modelling Language*, jika tidak ada diagram yang cocok untuk tujuan tersebut (Munawar, 2005, dalam Ade Kurniawan, 2012).

1. *Use Case diagram*

Use case diagram menjelaskan manfaat sistem jika dilihat menurut pandangan orang yang berada diluar sistem (*actor*). Diagram ini menunjukkan fungsionalitas suatu sistem atau *class* dan bagaimana sistem berinteraksi dengan dunia luar. *Use case diagram* dapat digunakan selama proses analisis untuk menangkap *requirements* sistem dan untuk memahami bagaimana sistem seharusnya bekerja. Selama tahap desain, *use case diagram* menetapkan perilaku sistem saat diimplementasikan. Dalam sebuah model mungkin terdapat satu atau beberapa *use case diagram*. (Munawar, 2005, dalam Ade Kurniawan, 2012).

Use case Diagram menyajikan interaksi antara *use case* dan aktor. Dimana, aktor dapat berupa orang, peralatan, atau sistem lain yang berinteraksi dengan sistem yang sedang dibangun. *Use case* menggambarkan fungsionalitas sistem atau persyaratan-persyaratan yang harus dipenuhi sistem dari pandangan pemakai.

2. *Class diagram*

Class diagram membantu dalam visualisasi struktur beberapa *class* dari suatu sistem dan merupakan tipe diagram yang paling banyak dipakai. *Class diagram* memperlihatkan hubungan antar *class* dan penjelasan detail tiap-tiap *class* di dalam model desain dari suatu sistem. Selama proses analisis, *class diagram* memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem. Selama tahap desain, *class diagram* berperan dalam menangkap struktur dari semua *class* yang membentuk arsitektur sistem yang dibuat. *Class diagram* juga merupakan fondasi untuk *component diagram* dan *deployment diagram*. Dalam sebuah model mungkin terdapat beberapa *class diagram* dengan spesifikasi tersendiri. (Munawar, 2005, dalam Ade Kurniawan, 2012).

3. *Sequence diagram*

Sequence diagram menjelaskan interaksi objek yang disusun dalam suatu urutan waktu. Diagram ini secara khusus berasosiasi dengan *use case*. *Sequence diagram* memperlihatkan tahap demi tahap apa yang seharusnya terjadi untuk menghasilkan sesuatu di dalam *use case*. Tipe diagram ini sebaiknya digunakan di awal tahap desain atau analisis karena kesederhanaannya dan mudah dimengerti. (Munawar, 2005, dalam Ade Kurniawan, 2012).

Sebuah *message* dari satu *participant* ke *participant* yang lain dan dari satu *lifeline* ke *lifeline* yang lain. Sebuah *participant* dapat mengirim sebuah *message* kepada dirinya sendiri. (Munawar, 2005, dalam Ade Kurniawan, 2012).

4. *Collaboration diagram*

Collaboration diagram melihat pada interaksi dan hubungan terstruktur antar objek. Tipe diagram ini menekankan pada hubungan antar objek, sedangkan *sequence diagram* menekankan pada urutan kejadian. Dalam satu *collaboration diagram* terdapat beberapa *object*, *link* dan *message*. *Collaboration diagram* digunakan sebagai alat untuk menggambarkan interaksi yang mengungkapkan keputusan mengenai perilaku sistem. (Munawar, 2005, dalam Ade Kurniawan, 2012).

5. *Activity diagram*

Activity diagram memodelkan alur kerja sebuah proses bisnis dan urutan aktivitas dalam suatu proses. Diagram ini sangat mirip dengan sebuah *flowchart* karena kita dapat memodelkan sebuah alur kerja dari satu aktifitas ke aktifitas lainnya atau dari satu aktifitas ke dalam keadaan sesaat (*state*). Seringkali bermanfaat bila kita membuat sebuah *activity diagram* terlebih dahulu dalam memodelkan sebuah proses untuk membantu kita dalam memahami proses secara keseluruhan. *Activity diagram* juga sangat berguna ketika kita ingin menggambarkan perilaku paralel atau menjelaskan bagaimana perilaku dalam berbagai *use case* berinteraksi. (Munawar, 2005, dalam Ade Kurniawan, 2012).

6. *Statechart diagram*

Kita dapat menggunakan *statechart diagram* untuk memodelkan perilaku dinamis satu *class* atau objek. *Statechart*

diagram memperlihatkan urutan keadaan sesaat (*state*) yang dilalui sebuah objek, kejadian yang menyebabkan sebuah transisi dari satu *state* atau aktifitas kepada yang lainnya, dan aksi yang menyebabkan perubahan satu *state* atau aktifitas. *Statechart diagram* khususnya digunakan untuk memodelkan tahap-tahap diskrit dari sebuah siklus hidup objek, sedangkan *activity diagram* paling cocok digunakan untuk memodelkan urutan aktifitas dalam suatu proses. (Munawar, 2005, dalam Ade Kurniawan, 2012).

7. *Component diagram*

Component diagram menggambarkan alokasi semua *class* dan objek ke dalam komponen-komponen dalam desain fisik sistem *software*. Diagram ini memperlihatkan pengaturan dan kebergantungan antara komponen-komponen *software*, seperti *source code*, *binary code*, dan komponen tereksekusi (*executable components*). Kita dapat membuat satu atau lebih komponen diagram untuk menggambarkan komponen dan paket atau menerangkan isi dari tiap-tiap paket komponen. (Munawar, 2005, dalam Ade Kurniawan, 2012).

8. *Deployment diagram*

Setiap model hanya memiliki satu *deployment diagram*. Diagram ini memperlihatkan pemetaan *software* kepada *hardware*. (Munawar, 2005, dalam Ade Kurniawan, 2012).

B. ***Eclipse Integrated Development Environment (IDE)***

Banyak orang lebih mengenal *EclipseIDE* sebagai *Integrated Development Environment* untuk bahasa pemrograman Java. Pada tahun 2014 *Eclipse IDE* mendominasi 65% pasar *Development Environment*. *Eclipse IDE* dapat ditambahkan dengan komponen-komponen *software* tambahan. *Eclipse* menyebut komponen-komponen ini dengan sebutan *plug-ins*. *Plug-ins* dapat dikelompokkan berdasarkan fitur-fiturnya.

Beberapa *open source projects* dan perusahaan mengembangkan *Eclipse IDE* atau membuat aplikasi mandiri (*Eclipse Rich Client Platform*) pada puncak *Eclipse Framework*. *Eclipse IDE* juga tersedia sebagai *Development Environment* untuk bahasa pemrograman lain seperti C, C++ sampai Lua, Python, Perl, dan PHP.

C. **Android Development Tools (ADT)**

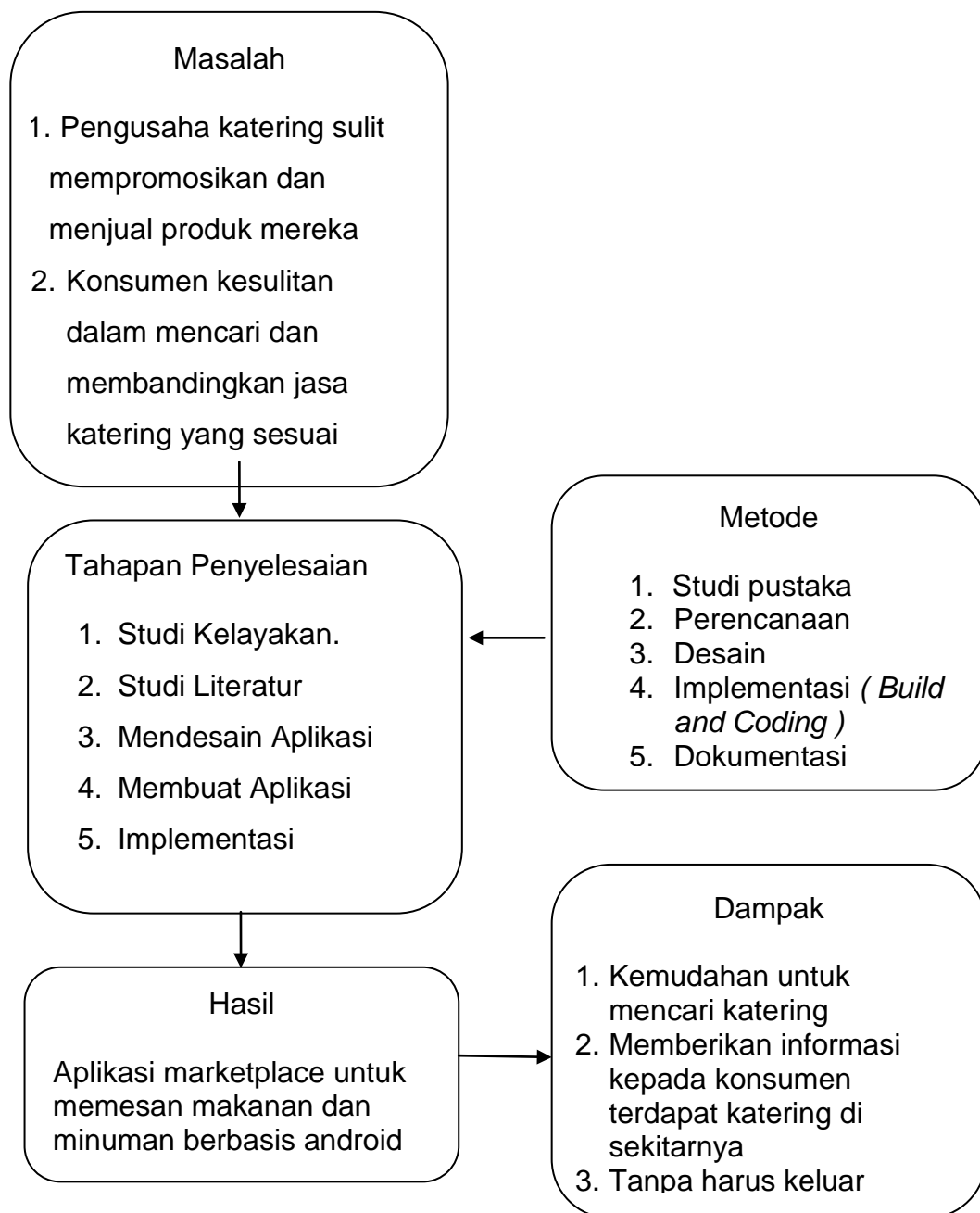
Android Development Tool (ADT) adalah sebuah *Plugin* yang berisi *source code*, paket dan *library* untuk mengembangkan proyek pembuatan aplikasi di dalam *platform* sistem operasi *Android*. Sebelum *Android Development Tool* dapat digunakan, terlebih dahulu kita harus mengintegrasikannya ke dalam *Eclipse IDE*.

2.2 Kerangka Pikir

Kerangka pikir adalah suatu peta hubungan antara variabel yang disusun dari berbagai teori yang telah dideskripsikan. Kerangka pikir menjelaskan secara garis besar alur logika berjalannya penelitian ini, dan dibuat berdasarkan rumusan masalah yang ada. Dalam sebuah wilayah di Jakarta dibutuhkan katering yang menjadi tempat untuk dikunjungi oleh karyawan pada saat istirahat untuk membeli makanan dan minuman, akan tetapi karena belum terdapat kantin di gedung perkantoran. Seperti halnya jika karyawan ingin membeli makanan dan minuman akan tetapi makanan dan minuman tersebut harus didapat dengan keluar gedung perkantoran terlebih dahulu sehingga memakan waktu cukup lama. Alangkah baiknya jika terdapat wadah komunitas bisnis interaktif secara elektronik dimana pedagang dapat melakukan registrasi kemudian meng-*upload* menu makanan dan minuman yang akan dijual olehnya untuk karyawan dan masyarakat pada umumnya. Sehingga karyawan dan masyarakat dapat melakukan pemesanan makanan dan minuman dengan berbagai pilihan dari pedagang yang telah melakukan registrasi pada katering *online* dan karyawan tidak perlu lagi keluar gedung pada jam istirahat serta mempercepat proses pemesanan makanan untuk masyarakat umum.

Agar aplikasi dapat dijalankan maka dibutuhkan perangkat yang bisa digunakan secara *mobile* atau bisa dipakai dimana saja yaitu *gadget* atau *smartphone*. Oleh karena itu jika terdapat aplikasi yang berjalan di perangkat *mobile* maka akan sangat memudahkan proses pemesanan makanan karena hampir setiap orang lebih sering menggunakan *gadget*. Karena selain pengoperasian yang lebih mudah, aplikasi *mobile* juga dapat mempercepat waktu pemesanan makanan maupun data, karena data dapat di-*input* kapan saja selama pengguna tersebut sedang menggunakan *gadget*-nya.

Sebagai bahan perbandingan dalam penelitian ini, penulis menggunakan penelitian lain yang relevan dengan judul yang penulis ambil, dengan tujuan untuk menyempurnakan ide dan teknik perancangan. Dengan membandingkan beberapa penelitian yang berbeda, akan muncul gagasan atau inspirasi yang tidak terduga dari pola pikir yang berbeda, yang akan membantu penulis dalam mengembangkan penelitian ini lebih jauh. Dapat di lihat dari gambar bagan kerangka berpikir.



Gambar 2.3 Bagan Kerangka Berpikir.

2.2.1 Penelitian Yang Relevan

Jenis penelitian yang digunakan sebagai pembanding disesuaikan pokok temanya yaitu aplikasi pemesanan makanan dan minuman. Penulis menggunakan jurnal penelitian yang terkait tema diatas dengan judul diantaranya:

A. Sistem Informasi Pemesanan Makanan dan Minuman pada Omaha Cafe and Resto Berbasis *Client Server* dengan Platform *Android*.

Penelitian ini dilakukan oleh Dian Galih Tegar, mahasiswa Sistem Informasi Universitas Dian Nuswantoro, Semarang. Dalam pembahasan dari jurnal penelitian, diambil garis besar tentang tujuan dari aplikasi yang dibuat, antara lain :

- 1) Merancang Aplikasi pemesanan makanan dan minuman berbasis *client server* dengan *platform* android pada Omaha Café and Resto.
- 2) Menghasilkan aplikasi yang dapat memberitahu pesanan pelanggan ke bagian dapur dan kasir.

Aplikasi ini bernama “Omaha”. Aplikasi ini mempunyai 2 antarmuka sistem yaitu *interface* sisi Admin (dapur dan kasir) dan *interface* sisi pelanggan. Secara keseluruhan, fitur yang dapat dilakukan oleh aplikasi ini diantaranya yaitu :

- 1) Dapat menampilkan informasi yang akurat mengenai semua makanan dan minuman yang tersedia di daftar menu.
- 2) Dapat mencatat menu yang dipesan, jumlah yang dipesan, nama pemesan dan lainnya.
- 3) Mengkonfirmasi pesanan kepada pelanggan.
- 4) Dapat meneruskan pesanan ke bagian terkait.

B. Pembuatan Aplikasi Layanan Pesan Antar Makanan pada Sistem Operasi *Android*.

Penelitian ini dilakukan oleh Muhammad Akbar, Kodrat Iman Satoto, dan R. Rizal Isnanto, mahasiswa Jurusan Teknik Elektro, Universitas Diponegoro, Semarang. Secara garis besar, aplikasi yang dibuat adalah aplikasi layanan pesan antar makanan, tetapi hanya tersedia satu tempat makanan atau resto saja.

2.2.2 Kerangka Studi Pembeding

Tabel 2.2 Studi Pembeding

No	Judul Penelitian	Sumber	Kelebihan	Kekurangan
1.	Sistem Informasi Pemesanan Makanan dan Minuman Pada <i>Omaha Cafe and Resto</i> Berbasis <i>Client Server</i> dengan <i>Platform Android</i>	Dian Galih Tegar, Jurnal Sistem Informasi Universitas Dian Nuswantoro, Semarang, 2013	<p>1. Aplikasi dapat mengatasi pemesanan makanan dan minuman dengan <i>platform Android</i> berbasis <i>client server</i>.</p> <p>2. Aplikasi <i>Mobile Android</i> ini bisa membantu pemesanan pelanggan ke bagian dapur dan kasir.</p>	1. Tidak ada pemberitahuan baik dengan sistem <i>alert</i> pada platform android atau pemberitahuan dengan sistem tampilan <i>display</i> layar monitor.
2.	Pembuatan Aplikasi Layanan Pesan Antar Makanan Pada Sistem	M. Akbar, Kodrat Iman Satoto, dan R. Rizal Isnanto. Jurnal	1. Pelanggan yang menggunakan Aplikasi ini dapat lebih mudah	1. Pada pelayanan pesan antar makanan ini hanya tersedia satu tempat makanan atau

	Operasi <i>Android</i>	Teknik Elektro, Universitas Diponegoro, 2014.	melakukan pesan serta dapat menghemat biaya.	satu resto saja
--	---------------------------	---	--	-----------------

BAB 3

METODE PENELITIAN

Pada bab ini akan dibahas mengenai pendekatan penelitian menggunakan metode pemodelan *Unified Modelling Language*. Pada bab ini juga akan dibahas tentang studi kelayakan aplikasi serta tempat dan waktu penelitian.

3.1 Pendekatan Penelitian

Pendekatan penelitian pada penulisan ini menggunakan Metodologi pengembangan sistem. Metodologi pengembangan sistem adalah metode dan prosedur yang digunakan untuk melakukan pengembangan perangkat lunak. Ada tiga jenis metodologi pengembangan sistem, yaitu :

1. *Structured Design*

Metodologi ini menggunakan pendekatan di dalam pengembangan sistem mengikuti tahapan siklus hidup sistem dan dibekali alat-alat dan teknik-teknik yang memadai. Metodologi yang menggunakan *structured design* meliputi :

- a. *Waterfall*, metodologi dengan sekuensial linier terdiri dari tahapan perencanaan sistem (rekayasa sistem), analisis kebutuhan, desain, penulisan program, pengujian, dan perawatan sistem.
- b. *Parallel*, metodologi dalam upaya untuk mengatasi interval waktu yang lama antara tahap analisis dan pengiriman sistem, caranya dengan membuat desain secara umum untuk seluruh sistem yang akan dilakukan dan kemudian membaginya dalam beberapa *sub-project* yang berbeda dan jalan secara bersama.

2. *Rapid Application Development (RAD)*

Rapid application development adalah sistem pemrograman yang memungkinkan pemrogram membuat program dengan

cepat. Secara umum, sistem *rapid application development* menyediakan sejumlah alat bantu untuk membuat antarmuka pengguna grafis *Graphical User Interface* (GUI) yang biasanya membutuhkan usaha dan waktu yang lama untuk membuatnya. Ada tiga metodologi dalam pengembangan sistem dengan *rapid application development*, yaitu :

- a. *Phased Development*.
- b. *Prototyping*.
- c. *Throw-Away Prototyping*.

Rapid application development menggunakan metode *prototyping* dan teknik terstruktur lainnya untuk menentukan kebutuhan *user* dan perancangan sistem informasi.

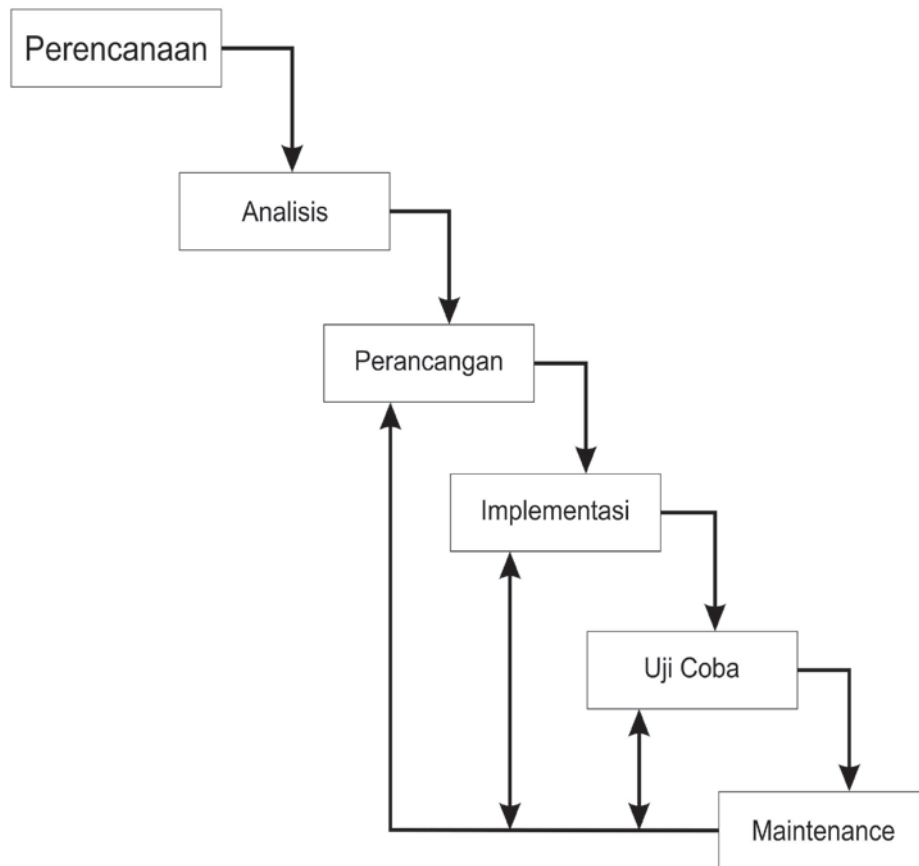
3. *Agile Development*

Metodologi pengembangan sistem dengan kategori *agile development* lebih fokus dalam pelurusan penggunaan *System Development Life Cycle* (SDLC) dengan mengeliminasi dalam hal pemodelan dan dokumentasi, sehingga tidak menghabiskan waktu dalam proses pengembangan sistem. Untuk mencapainya, metodologi ini menggunakan :

- a. *Extreme Programming*, dan
- b. *Scrum*.

3.1.1 **Structured Design Model Waterfall**

Pembuatan Aplikasi *marketplace* catering yang dibuat pada platform *Android* ini dilakukan secara bertahap. Penulis menggunakan model *Structured Design waterfall*, yaitu sebuah model proses pengembangan perangkat lunak yang setiap tahapannya digambarkan mengalir ke bawah. Tahapan yang dilalui penulis meliputi perencanaan, analisa, perancangan, implementasi, uji coba dan maintenance, disajikan dalam blok diagram yang ditunjukkan oleh gambar 3.1:



Gambar 3.1 Bagan Model *Waterfall*

Keunggulan menggunakan teknik *waterfall* adalah proses menjadi teratur, estimasi proses menjadi lebih baik dan jadwal menjadi lebih menentu tetapi sifatnya masih kaku, sehingga susah melakukan di tengah proses serta membutuhkan daftar kebutuhan yang lengkap di awal, akan tetapi jarang konsumen bisa memberikan kebutuhan secara lengkap di awal.

Berikut ini merupakan deskripsi dari tahapan – tahapan yang dilakukan penulis dalam pembuatan aplikasi *e-marketplace* catering ini, yaitu :

A. Perencanaan

Tahapan pertama dari pembuatan aplikasi ini yaitu mengumpulkan informasi atau data yang diperlukan dan berkaitan dengan materi yang dibahas, diantaranya adalah:

- 1) Bahasa pemrograman Java
- 2) *Eclipse* dan *Android Development Tools*
- 3) Sistem operasi Android
- 4) Pemodelan *Unified Modelling Language*(UML)
- 5) *Database MySQL*

B. Analisis

Tahapan ini merupakan proses analisa kebutuhan, yaitu merencanakan bagaimana aplikasi akan dibuat, serta perangkat lunak dan perangkat keras apa saja yang dibutuhkan untuk dapat membuat aplikasi *e-marketplace* catering ini. Perangkat keras yang digunakan oleh penulis adalah seperangkat laptop dan *smartphone android* yang berperan sebagai tempat melakukan *testing* aplikasi. Perangkat lunak yang digunakan meliputi sistem operasi laptop dan *smartphone* serta *integrated development environment (compiler)*.

C. Perancangan

Pada tahapan ini perancangan ini meliputi pembuatan desain aplikasi serta UML Diagram dari aplikasi yang akan dibuat, sebagai bahan acuan untuk membuat aplikasi yang mudah untuk dioperasikan oleh pengguna. Selain itu, penulis juga melakukan perancangan halaman atau yang biasa disebut *story board* yang diperlukan untuk dasar pembuatan aplikasi yang dimaksud, sehingga tampilan antarmuka aplikasi menjadi lebih terorganisir dan efektif.

D. Implementasi

Pada tahapan implementasi ini dilakukan pembuatan aplikasi secara keseluruhan yang meliputi proses pengetikan kode program (*coding*) pada Eclipse Kemudian melakukan proses uji coba aplikasi pada *smartphone Android*, apakah sudah benar-benar dapat dioperasikan sebagaimana fungsinya.

E. Uji coba

Aplikasi yang telah jadi harus diuji coba setiap fungsi-fungsinya agar dapat diketahui permasalahannya dan juga sesuai dengan kebutuhan yang diperlukan.

F. Maintenance

Pada tahap *maintenance* dilakukan pemeliharaan yaitu perawatan aplikasi agar dapat berjalan dengan baik dan termasuk juga di dalamnya proses pengembangan aplikasi agar menjadi lebih baik lagi sesuai dengan kebutuhan.

3.2 Instrumen Penelitian

Pada sub-bab ini akan dijelaskan mengenai instrumen penelitian yang digunakan. Instrumen penelitian yang dilakukan dalam penulisan ini yaitu menggunakan studi kelayakan dan metode pengumpulan data melalui observasi dan kuesioner.

3.2.1 Studi kelayakan

Pembahasan studi kelayakan dalam pembuatan aplikasi *e-marketplace* catering menggunakan rumusan masalah sebagai acuan kelayakan. Kelayakan dinilai dari dua segi pembahasan yaitu, segi kelayakan teknis dan kelayakan ekonomi.

A. Kelayakan teknis

Kelayakan teknis dinilai dari bagaimana cara aplikasi memberikan kemudahan dalam membantu pengguna untuk melakukan proses pemesanan makanan dan minuman.

Pengguna yang akan melakukan pemesanan dengan berbagai menu kini dapat melakukan pemesanan melalui *smartphone* mereka secara sederhana dengan aplikasi *e-marketplace* catering. Dari segi teknis, lebih mudah dibanding melakukan pemesanan dengan cara mendatangi tempat makan secara langsung, karena pengguna hanya harus melakukan pemesanan dari dalam gedung kampus, memilih makanan yang diinginkan, memberitahu lokasi. Selanjutnya makanan yang telah dipesan diproses oleh pengelola catering *online*, dan dikirim sesuai dengan alamat yang telah ditentukan pengguna.

B. Kelayakan ekonomi

Di dalam Aplikasi *e-marketplace* catering tersedia daftar kategori menu makanan dan minuman dari pengelola catering yang sebelumnya telah melakukan registrasi. Pengguna dapat memilih kategori apa yang mereka inginkan, kemudian melakukan pemesanan. Dengan era teknologi *smartphone* pada saat ini, pengguna tidak perlu keluar gedung hanya untuk membeli makanan dan pengguna dapat tahu dengan jelas makanan dan minuman apa yang akan mereka pesan. Pengguna tidak perlu pergi dan membuang biaya bensin/tenaga untuk berkeliling mencari tempat makan ketika jam istirahat, atau menghabiskan biaya pulsa untuk memesan makanan melalui telepon.

3.2.2 Metodologi pengumpulan data

Teknik pengumpulan data yang dipilih yaitu dengan cara pendekatan kualitatif. Pendekatan kualitatif adalah penelitian yang

digunakan untuk meneliti pada kondisi objek alamiah dimana peneliti merupakan instrumen kunci (menurut Sugiyono : 2005 dalam Haryanto).

Berikut teknik pengumpulan data dengan menggunakan pendekatan kualitatif, yang diantaranya adalah :

A. Wawancara

Untuk mendapatkan informasi secara lengkap maka dilakukan suatu metode tanya jawab atau *interview* kepada beberapa pihak atau bagian produsen catering yang terlibat langsung terkait semua kegiatan yang berhubungan dengan proses pemesanan/pembelian makanan mulai dari pengelola catering hingga konsumen.

B. Observasi

Didalam artian penelitian observasi adalah mengadakan pengamatan secara langsung, observasi dapat dilakukan dengan tes, kuesioner, ragam gambar, dan rekaman suara. Pedoman observasi berisi sebuah daftar jenis kegiatan yang mungkin timbul dan akan diamati.

C. Studi Pustaka

Teknik pengumpulan data terakhir dengan mempelajari informasi dan mengumpulkan data dengan melihat sumber kepustakaan seperti situs *web*, jurnal ilmiah, skripsi, buku-buku, dan sebagainya yang tentunya dapat dipertanggung jawabkan untuk dijadikan bahan referensi dalam pembuatan aplikasi.

3.3 Tempat dan Waktu Penelitian

Dalam mengerjakan penulisan ini, penulis melakukan penelitian langsung di tempat yang sesuai dengan pokok pembahasan pada judul penulisan ini yaitu di daerah, Cipete, Lubang Buaya dan Rawa Belong, Jakarta Selatan, Jakarta Timur, Jakarta Pusat dan Jakarta Barat.

BAB 4

ANALISIS STRUKTUR PROGRAM APLIKASI

Pada bab ini akan dibahas mengenai proses bisnis aplikasi, program *flowchart*, dan tujuan perancangan aplikasi. Analisis struktur program aplikasi menggunakan metode berorientasi objek dan *Unified Modelling Language* sebagai alat bantu (*tools*) pemodelan. Pada metode analisis berorientasi objek dengan pemodelan *Unified Modelling Language* dalam aplikasi ini, digunakan empat diagram yaitu, diagram *activity*, diagram *use case*, diagram *class*, dan diagram *sequence*.

4.1 Proses Bisnis Program Aplikasi.

Pada bagian ini, akan dijabarkan tentang sistematika proses program aplikasi. Penjelasan proses bisnis aplikasi akan menggambarkan secara umum bagaimana aplikasi berjalan mengacu pada batasan masalah yang ada.

Aplikasi *E-Marketplace Catering* ini terdiri dari dua kategori, yaitu *login* untuk anggota dan *login* untuk pengelola. Halaman anggota memungkinkan anggota untuk melihat produk dari Tiap Jasa *Catering* yang ada, melihat daftar *Catering*, daftar menu pada masing-masing *Catering*, kategori, deskripsi, harga, gambar, dan keterangan. Anggota juga dapat memesan makanan dan minuman yang dipilih dan melihat detail *order* makanan atau minuman yang telah dipesan.

Untuk masuk ke dalam aplikasi, *user* harus melakukan registrasi mengisi nama, alamat, nomor telepon, *email*, *username*, dan *password*. Setelah terdaftar *user* dapat *login* terlebih dahulu, dengan mengisi *username* dan *password*. Setelah *user* berhasil *login*, maka *user* telah menjadi anggota *E-Marketplace Catering*. Anggota akan melihat tampilan menu utama yang terdiri dari enam tombol, tombol tersebut adalah “*Search*”, “*Profil*”, “*Daftar E-Marketplace Catering*”, “*Order*”, “*About*”, dan tombol “*Tagihan*”. Pada tombol “*Search*”, anggota dapat mencari menu makanan dan minuman yang akan

dipesan tanpa harus menekan tombol daftar *E-Marketplace Catering*. Pada tombol “Daftar *E-Marketplace Catering*”, anggota akan melihat tampilan daftar *E-Marketplace Catering*, melihat beberapa menu makanan yang ada pada setiap *E-Marketplace Catering*, setelah memilih makanan atau minuman yang diinginkan anggota dapat melakukan pemesanan. Tombol “Profil” berfungsi untuk menampilkan kode anggota, nama anggota, alamat, nomor telepon, *email*, *username*, dan *password*. Anggota juga dapat mengubah biodata sesuai keinginan anggota. Pada tombol “Order” anggota akan ditunjukkan kepada halaman produk yang telah dipilih dan dapat mengubah pilihan makanan atau minuman apabila anggota belum mengkonfirmasi tagihan. Pada tombol “About” anggota dapat melihat tentang aplikasi. Sedangkan pada tombol “Tagihan”, anggota dapat melakukan konfirmasi pesannya setelah mengetahui total tagihan, total *item* yang telah dipilih, serta mengisi tujuan pengiriman.

Pada halaman utama pengelola terdapat 7 tombol utama, yaitu *home*, profil, anggota, produk, *order*, arsip, dan *logout*.

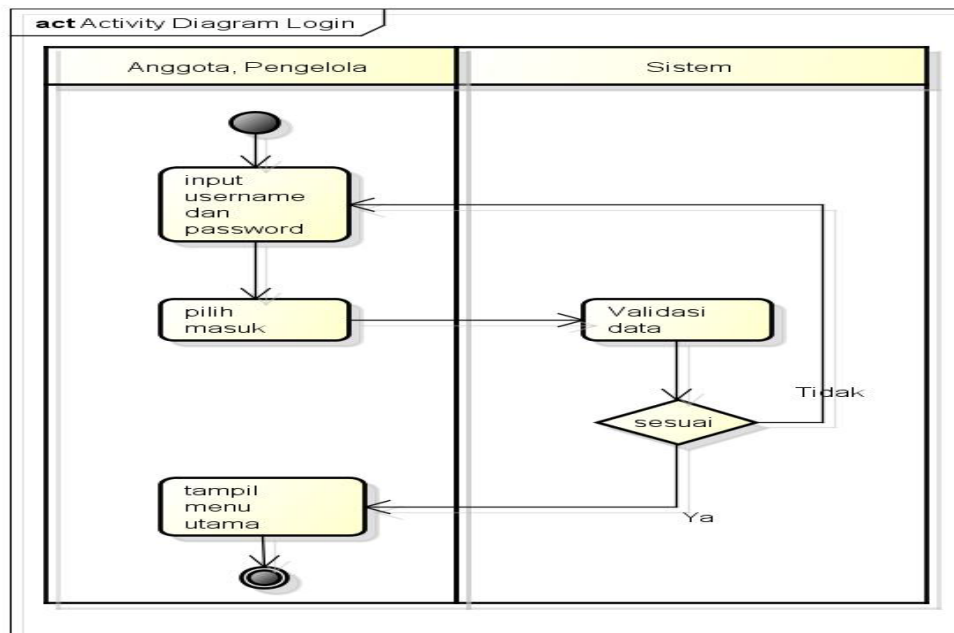
4.2 Activity Diagram.

Activity diagram menggambarkan logika prosedural, proses bisnis dan aliran aktifitas dalam aplikasi yang sedang dirancang. *Activity diagram* akan menggambarkan bagaimana proses dimulai, *decision* yang mungkin terjadi dan akhir aliran aktifitas. *Activity diagram* yang menjelaskan aktifitas aplikasi *E-Marketplace Catering*.

4.2.1 Activity Diagram Login.

Pada *Activity Diagram login* ini aktor yang melakukan adalah anggota dan pengelola. *Login* diperuntukkan bagi aktor yang akan masuk kedalam sistem dengan menginputkan *username* dan *password* kemudian pilih masuk. Selanjutnya sistem akan

memvalidasi data jika *username* dan *password* benar maka akan masuk ke dalam sistem jika *username* dan *password* salah maka akan kembali ke tampilan awal *login* terlihat pada gambar 4.1 *Activity Diagram Login*.

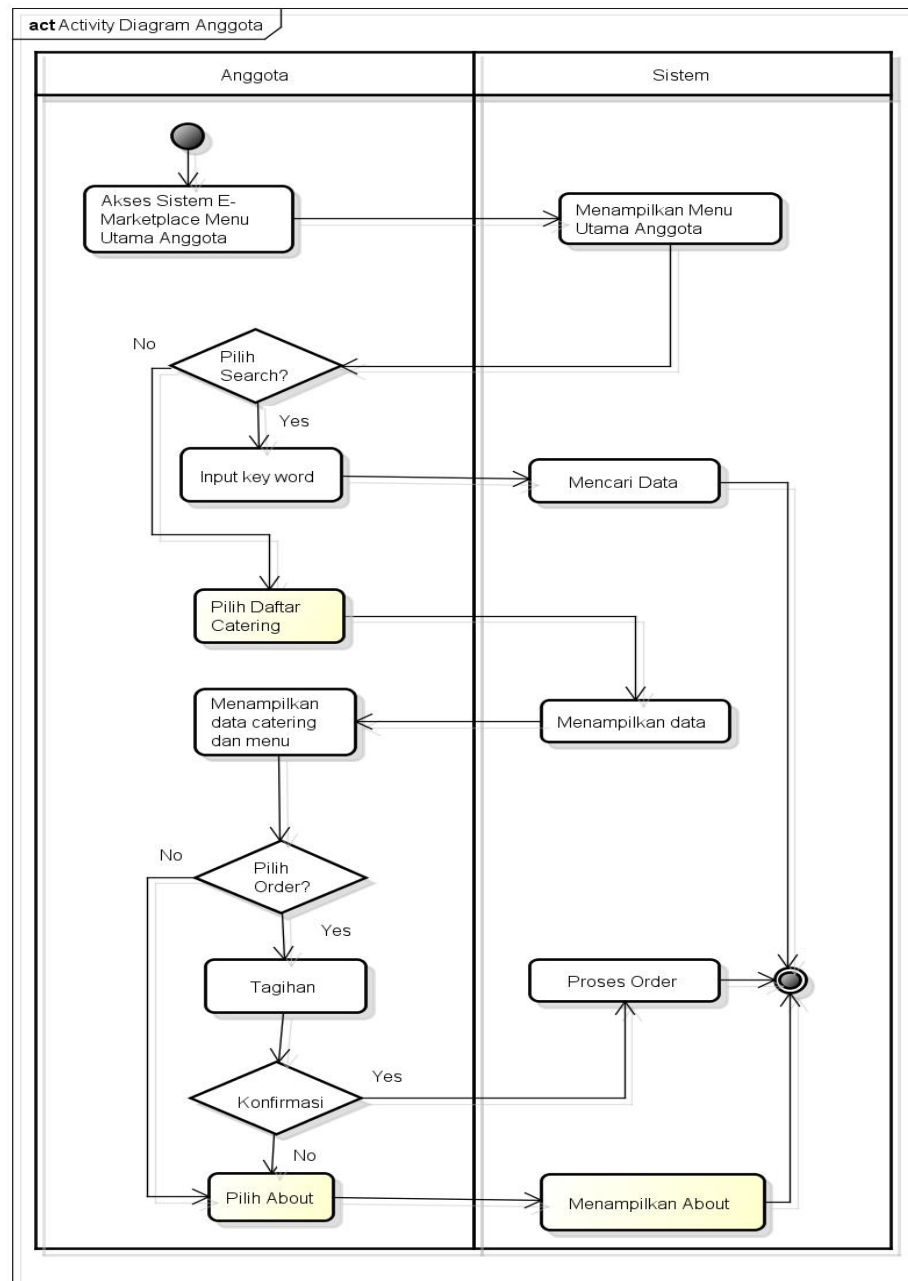


Gambar 4.11 *Activity Diagram Login*.

4.2.2 *Activity Diagram Anggota*.

Pada *activity* diagram menu utama anggota ini bisa di akses jika anggota telah berhasil *login*. Di dalam menu ini terdapat pilihan *Search* untuk mencari data *catering*. Aktor diminta menginputkan kata kunci yang ingin di cari selanjutnya sistem akan mencari data. Pilihan berikutnya adalah daftar *catering* jika memilih menu ini anggota bisa melihat daftar *catering* beserta produk nya selanjutnya anggota bisa langsung *order* dengan memilih *order*. Setelah *order* selesai maka akan tampil tagihan untuk selanjutnya anggota diminta konfirmasi *order* nya jika "Ya" maka sistem akan memproses *order*, Jika "tidak"

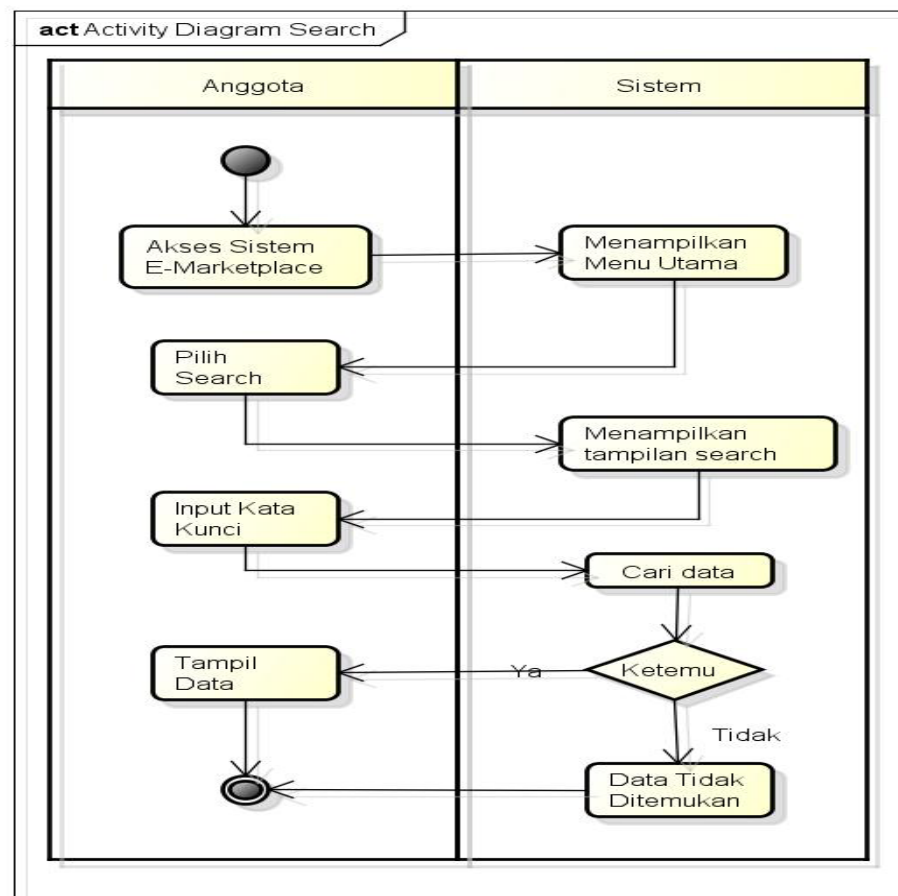
maka anggota bisa memilih menu selanjutnya yaitu menu *About*. Jika memilih menu *About* maka sistem akan menampilkan profil dari anggota. Jika tidak maka keluar dari sistem terlihat pada gambar 4.2 diagram anggota.



Gambar 4.2 Activity Diagram Anggota.

4.2.3 Activity Diagram Search.

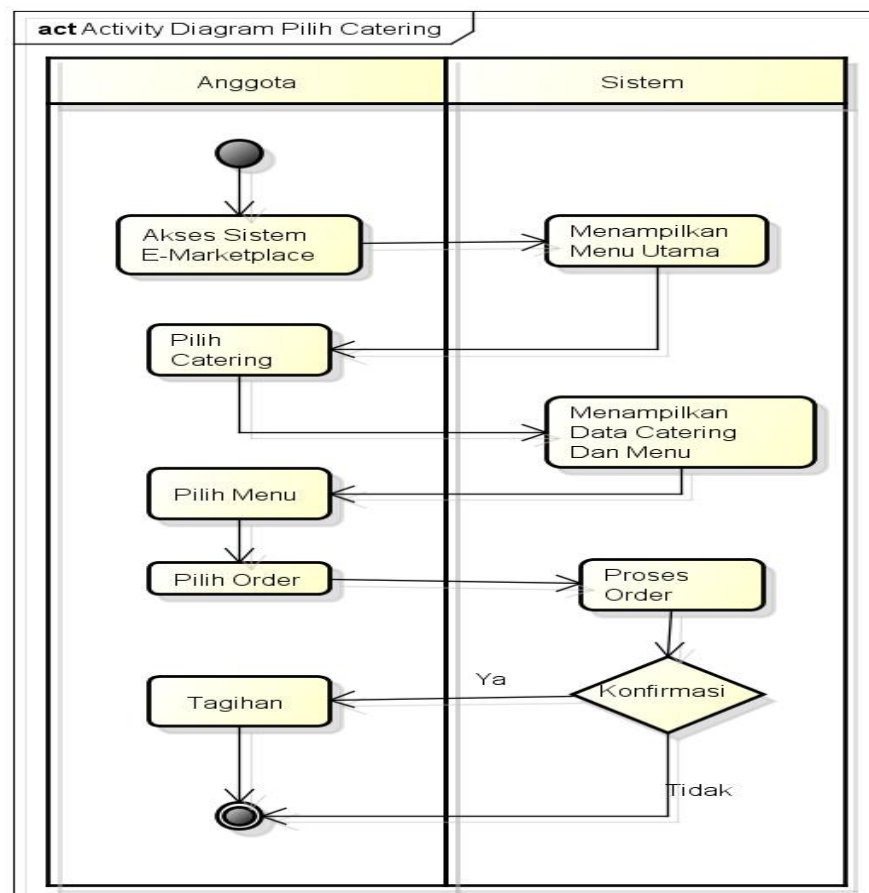
Untuk Memilih Menu *search* maka anggota harus mengakses menu utama terlebih dahulu, pada *activity diagram search* aktor diminta menginputkan kata kunci yang akan di cari selanjutnya sistem akan melakukan proses pencarian data berdasarkan kata kunci yang dimasukkan jika ketemu maka data akan di tampilkan jika tidak maka muncul pesan data tidak di temukan terlihat pada gambar 4.3 diagram *search*.



Gambar 4.3 Activity Diagram Search.

4.2.4 Activity Diagram Catering.

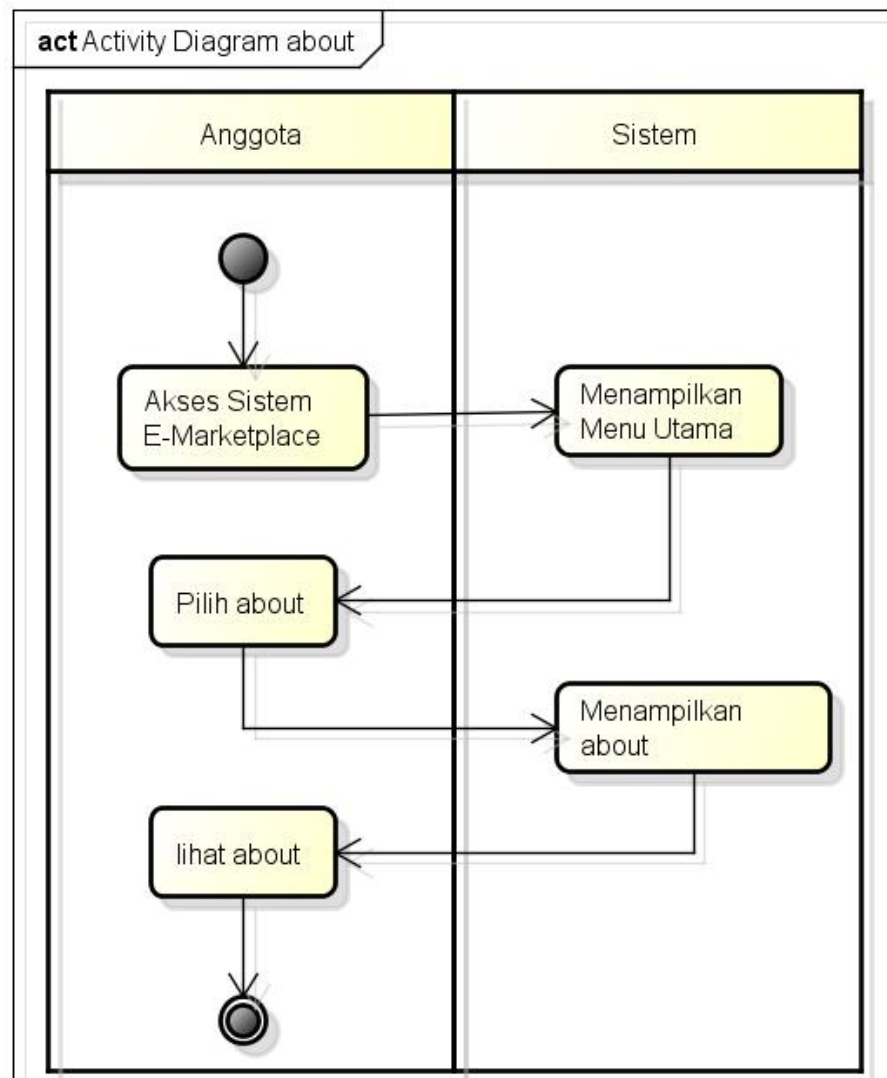
Untuk Memilih Menu data *catering* maka anggota harus mengakses menu utama terlebih dahulu, pada *activity* diagram menu *catering* aktor memilih daftar *catering* selanjutnya sistem akan menampilkan data *catering* beserta produknya selanjutnya anggota bisa langsung *order* dengan memilih *order*. Setelah *order* selesai maka akan tampil tagihan untuk selanjutnya anggota diminta konfirmasi *order* nya jika "Ya" maka sistem akan memproses *order*. Jika "tidak" maka *activity* selesai terlihat pada gambar 4.4 diagram *Catering*.



Gambar 4.4 Activity Diagram Catering.

4.2.5 Activity Diagram About.

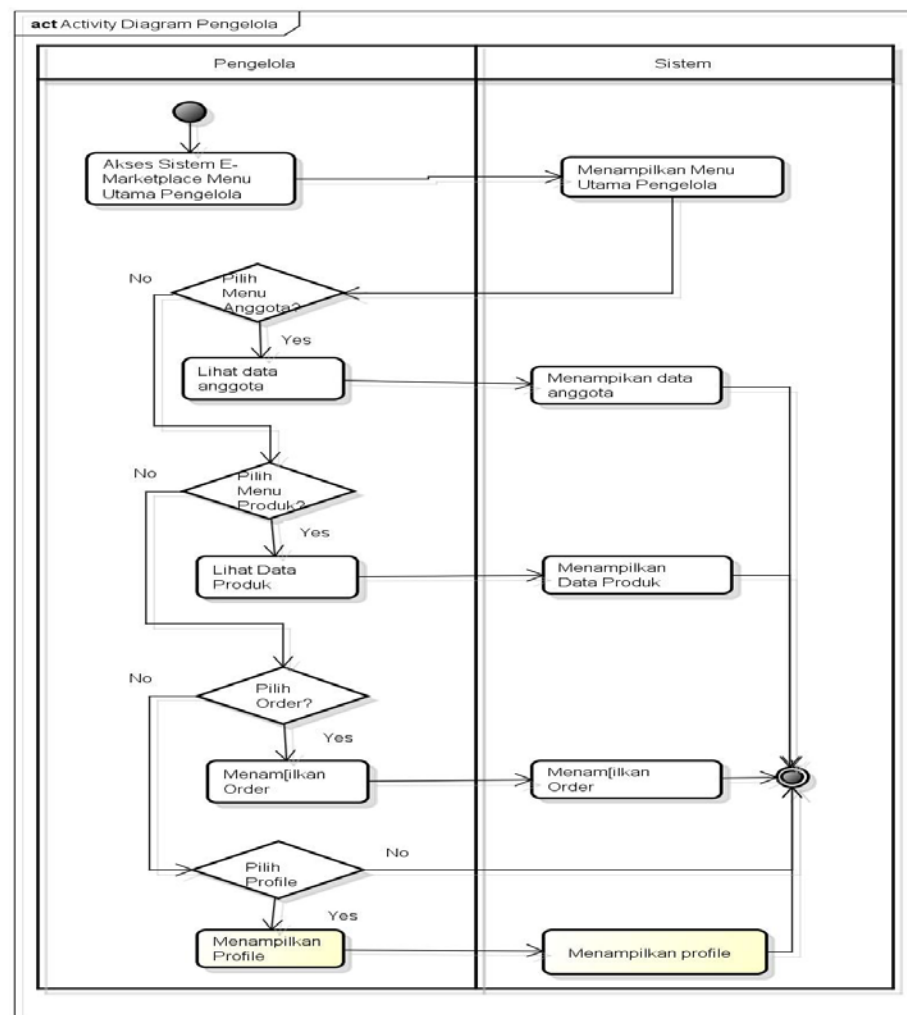
Untuk Memilih Menu data *About* maka anggota harus mengakses menu utama terlebih dahulu, pada *activity* diagram menu *About* aktor memilih *About* selanjutnya sistem akan menampilkan data *About* terlihat pada gambar 4.5 diagram *About*.



Gambar 4.5 Activity Diagram About.

4.2.6 Activity Diagram Pengelola.

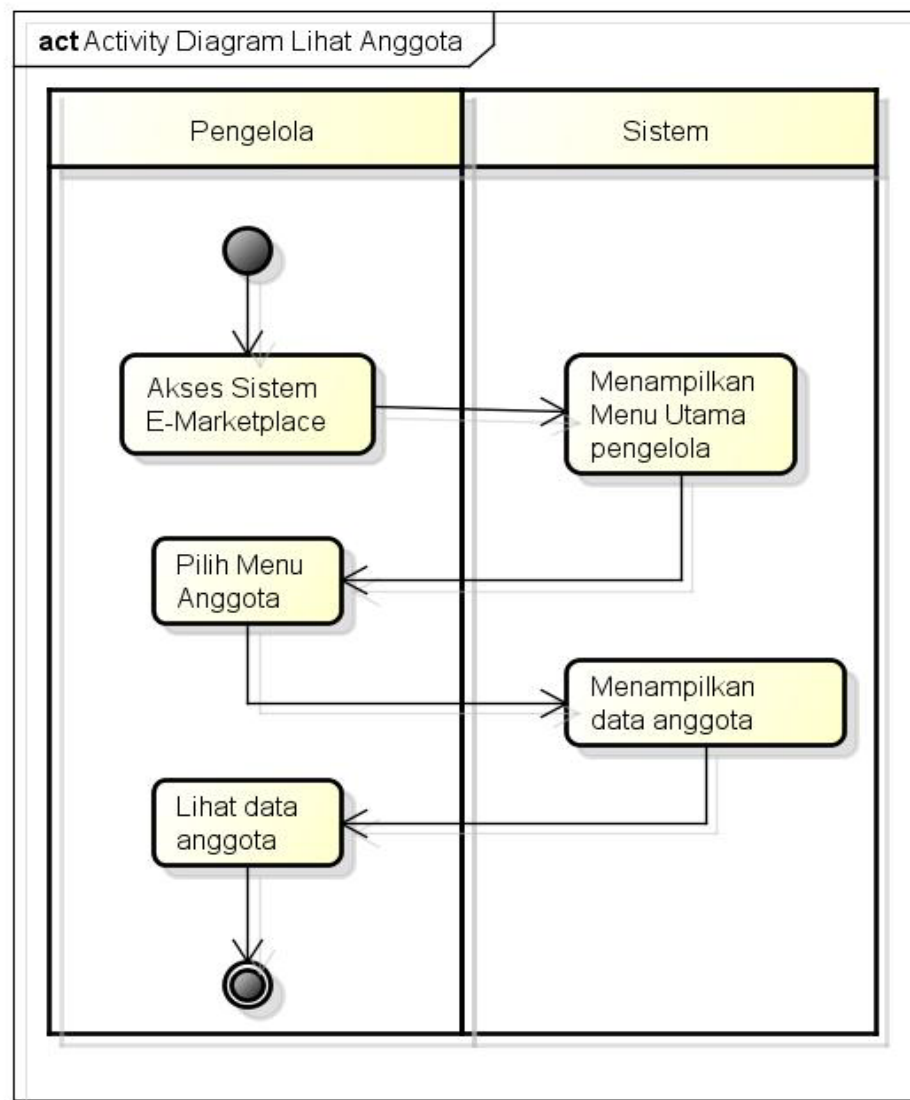
Pada *activity* diagram menu utama Pengelola ini bisa di akses jika pengelola telah berhasil *login*. Di dalam menu ini terdapat pilihan 4 pilihan menu yaitu Menu Lihat data anggota untuk menampilkan data anggota, menu Lihat data produk untuk melihat data produk, Menu *Order* untuk melihat data *order* yang masuk dan Menu profile untuk melihat *Profile catering*. Jika "tidak" maka keluar dari sistem terlihat pada gambar 4.6 diagram pengelola.



Gambar 4.6 Activity Diagram Pengelola.

4.2.7 Activity Diagram Lihat Anggota.

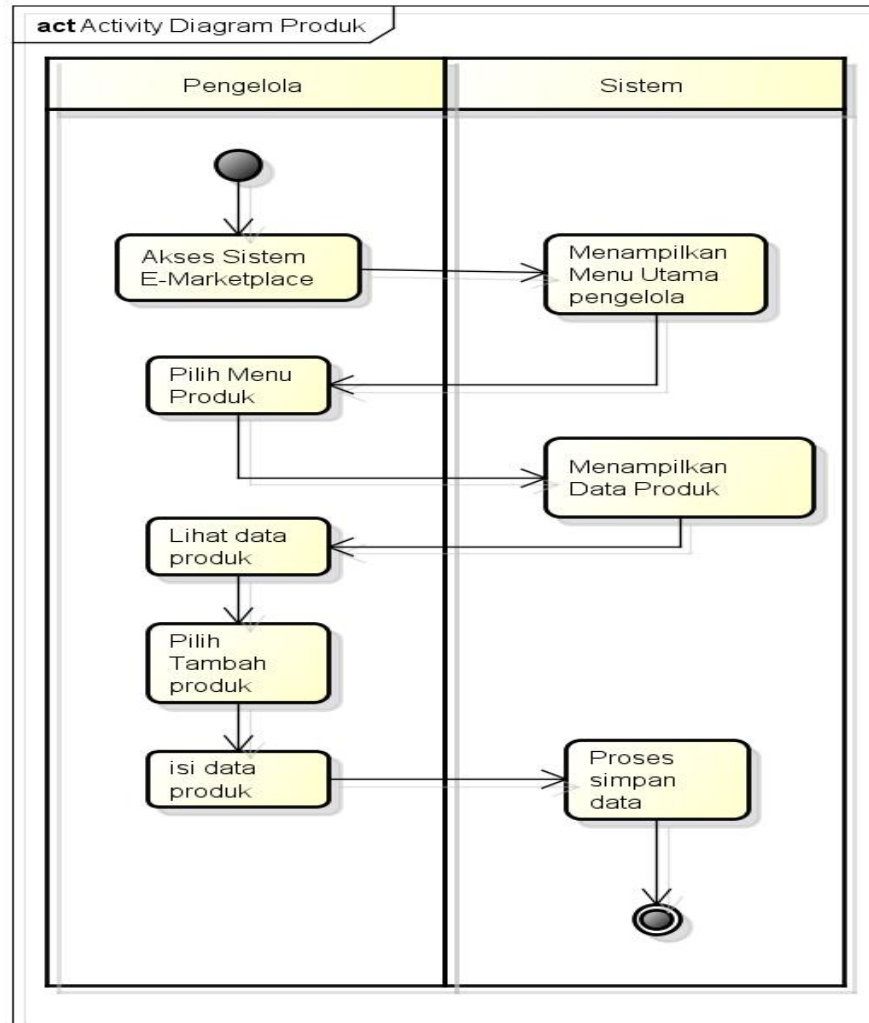
Pada *activity* diagram lihat anggota ini pengelola harus mengakses menu utama pengelola terlebih dahulu. Aktor memilih menu lihat anggota selanjutnya sistem akan menampilkan data anggota pengelola bisa melihat data anggota terlihat pada gambar 4.7 diagram anggota.



Gambar 4.7 Activity Diagram Anggota.

4.2.8 Activity Diagram Produk.

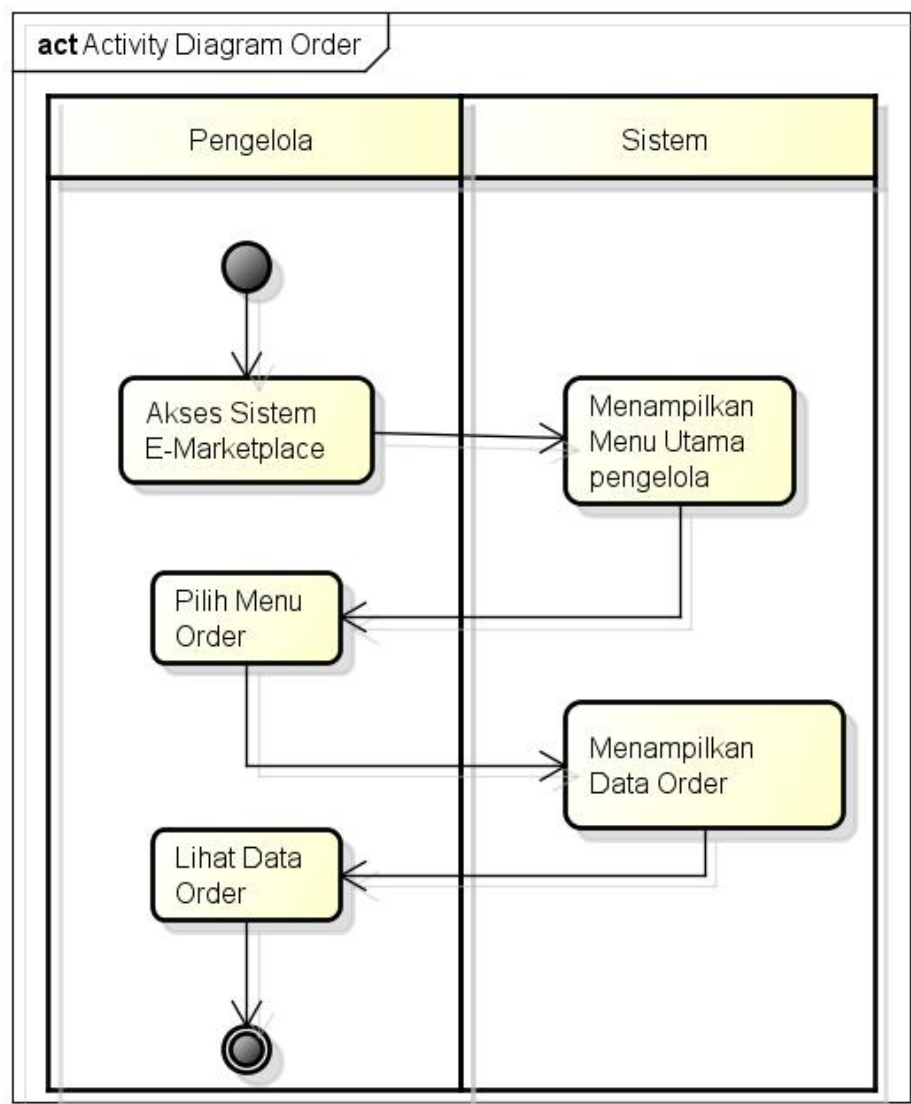
Pada *activity* diagram produk ini pengelola harus mengakses menu utama pengelola terlebih dahulu. Aktor memilih menu produk selanjutnya sistem akan menampilkan data produk pengelola bisa melihat data produk dan menambah data produk dengan mengisi data produk selanjutnya sistem akan menyimpan data produk yang telah di masukkan oleh aktor terlihat pada gambar 4.8 diagram produk.



Gambar 4.8 Activity Diagram Produk.

4.2.9 Activity Diagram Order.

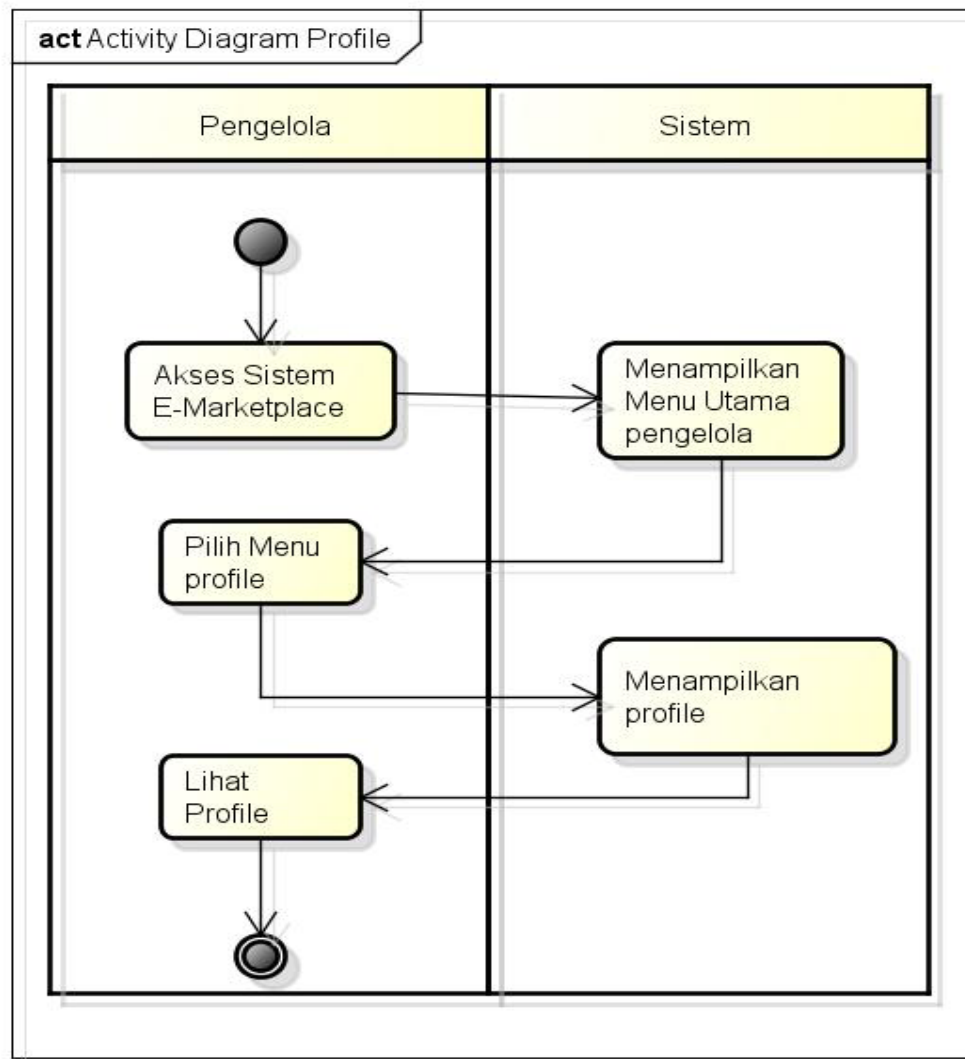
Pada *activity* diagram *order* ini pengelola harus mengakses menu utama pengelola terlebih dahulu. Aktor memilih menu *order* selanjutnya sistem akan menampilkan data *order* dari anggota pengelola bisa melihat data *order* untuk menindaklanjutinya aktor terlihat pada gambar 4.9 diagram *order*.



Gambar 4.9 Activity Diagram Order.

4.2.10 Activity Diagram Profile.

Pada *activity* diagram lihat *Profile* ini pengelola harus mengakses menu utama pengelola terlebih dahulu. Aktor memilih menu *Profile* selanjutnya sistem akan menampilkan *Profile* dari pengelola terlihat pada gambar 4.10 diagram *Profile*.



Gambar 4.10 Activity Diagram Profile.

4.3 Identifikasi Aktor.

Aktor dalam *Unified Modelling Language* adalah entitas manusia atau mesin (pengguna, *administrator*, pengunjung dan lain sebagainya) yang berinteraksi langsung dengan sistem untuk melakukan pekerjaan tertentu. Pada aplikasi “*E-Marketplace Catering*” terdapat satu aktor yaitu anggota.

4.3.1 Use Case Description.

Use Case Description adalah cara menggambarkan perilaku setiap *use case* dalam *use case diagram*. Sebuah *use case* merepresentasikan sebuah interaksi antara aktor dengan sistem, *Use Case* merupakan sebuah pekerjaan tertentu. Dari analisis terhadap kebutuhan fitur-fitur aplikasi dan identifikasi aktor, maka bisa didefinisikan terdapat 10 *use case description*. Penjelasan tiap-tiap *use case* dijelaskan pada tabel 4.1 sampai dengan 4.10.

Tabel 4.1 *Use case description “login”*.

Nama Use case	<i>Login</i>
Aktor	Anggota dan Pengelola
Deskripsi	Syarat masuk ke dalam <i>system</i> aplikasi
Normal course	<ol style="list-style-type: none"> 1. Anggota membuka aplikasi 2. Mengisikan <i>username</i> dan <i>password</i> 3. Pengguna masuk ke dalam sistem aplikasi

Tabel 4.2 *Use case description “Search”*.

Nama Use case	<i>Search</i>
Aktor	Anggota
Deskripsi	<ol style="list-style-type: none"> 1. Pencarian menu makanan dan minuman

Normal course	<ol style="list-style-type: none"> 1. Anggota menyetik nama menu pada tombol <i>search</i> 2. Anggota menemukan makanan dan minuman yang diinginkan apabila makanan atau minuman tersebut tersedia di <i>E-Marketplace Catering</i>
----------------------	---

Tabel 4.3 *Use case description* "Daftar *E-Marketplace Catering*".

Nama Use case	Daftar <i>E-Marketplace Catering</i>
Aktor	Anggota
Deskripsi	Melihat daftar <i>E-Marketplace Catering</i> di Jakarta
Normal course	<ol style="list-style-type: none"> 1. Anggota menekan tombol "Daftar <i>E-Marketplace Catering</i>" 2. Akan muncul tampilan daftar <i>E-Marketplace Catering</i> untuk melihat menu makanan dan minuman 3. Anggota dapat melakukan pemesanan makanan dan minuman setelah memilih menu makanan atau

Tabel 4.4 *Use case description* "Profil".

Nama Use case	Profil
Aktor	Anggota
Deskripsi	Mengubah biodata anggota
Normal course	<ol style="list-style-type: none"> 1. Anggota menekan tombol "Profil" 2. Mengganti biodata anggota 3. Anggota menyimpan data

Tabel 4.5 *Use case description order “Anggota”*.

Nama Use case	<i>Order</i>
Aktor	Anggota
Deskripsi	Melihat menu yang telah dipesan
Normal course	<ol style="list-style-type: none"> 1. Anggota menekan tombol “<i>Order</i>” 2. Anggota dapat melihat menu yang telah dipesan

Tabel 4.6 *Use case description “About”*.

Nama Use case	<i>About</i>
Aktor	Anggota
Deskripsi	Melihat info tentang aplikasi <i>E-Marketplace Catering</i>
Normal course	<ol style="list-style-type: none"> 1. Anggota menekan tombol “<i>About</i>” 2. Akan muncul informasi mengenai aplikasi <i>E-Marketplace Catering</i>

Tabel 4.7 *Use case description “Tagihan”*.

Nama Use case	Tagihan
Aktor	Anggota
Deskripsi	Melihat tagihan harga berdasarkan pesanan
Normal course	<ol style="list-style-type: none"> 1. Anggota menekan tombol “Tagihan” 2. Akan muncul informasi mengenai total tagihan, total <i>item</i>, dan dapat menulis tujuan pengiriman 3. Anggota dapat memberikan konfirmasi agar makanan dan minuman dapat diproses

Tabel 4.8 *Use case Description order* “Pengelola”.

Nama Use case	<i>Order</i>
Aktor	Pengelola
Deskripsi	Melihat daftar pesanan
Normal course	<ol style="list-style-type: none"> 1. Pengelola menekan tombol “<i>order</i>”. 2. Akan muncul tampilan daftar pesanan apabila pengelola mendapatkan <i>order</i> dari anggota.

Tabel 4.9 *Use case description* “Produk”.

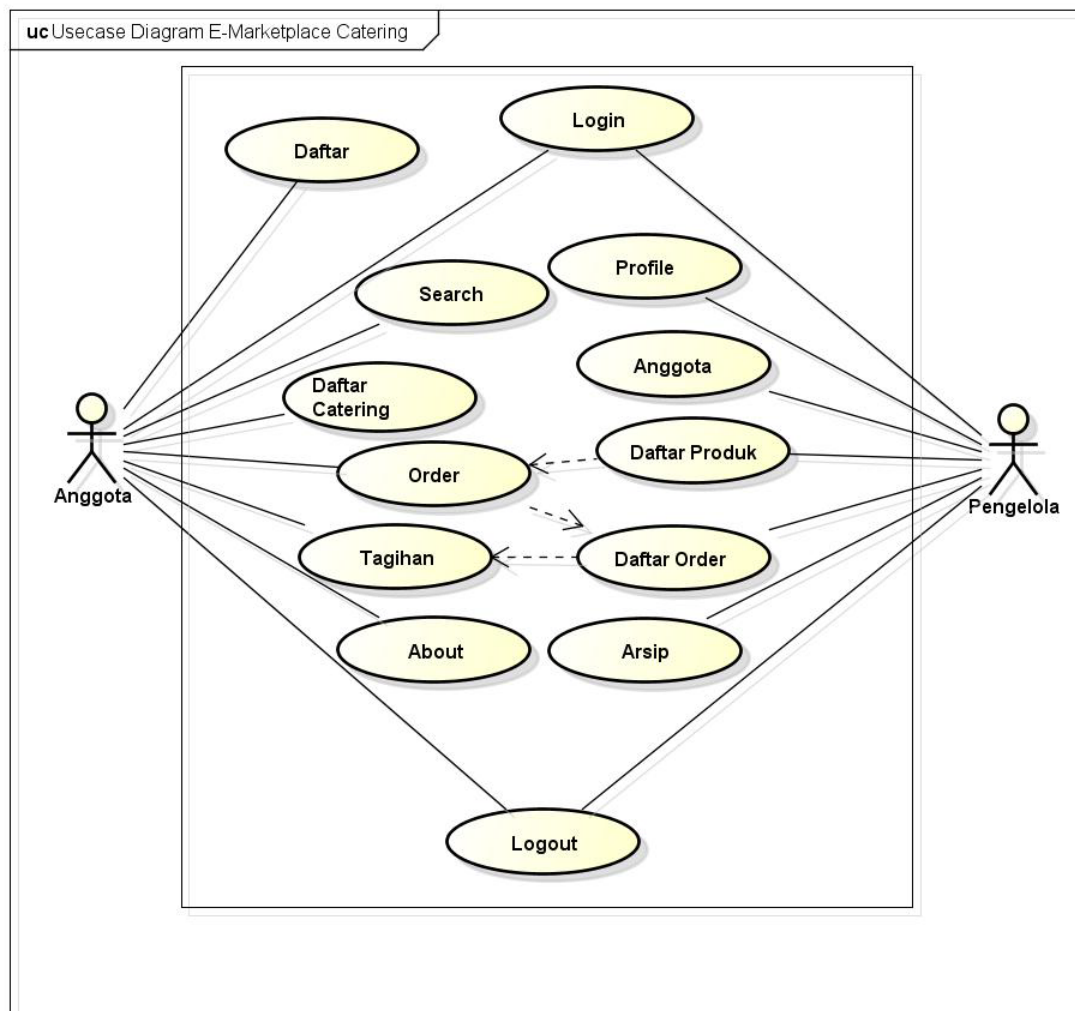
Nama Use case	Produk
Aktor	Pengelola
Deskripsi	Menginput produk menu makanan dan minuman
Normal course	<ol style="list-style-type: none"> 1. Pengelola menekan tombol “<i>Input Menu</i>” 2. Pengguna dapat memasukkan menu makanan dan minuman untuk di<i>order</i> oleh anggota

Tabel 4.10 *Use case description* profil “Pengelola”.

Nama Use case	Profil Pengelola
Aktor	Pengelola
Deskripsi	Mengubah biodata pengelola
Normal course	<ol style="list-style-type: none"> 1. Pengelola menekan tombol “<i>Profil</i>” 2. Pengelola dapat mengganti data yang ada

4.3.2 Use Case Diagram

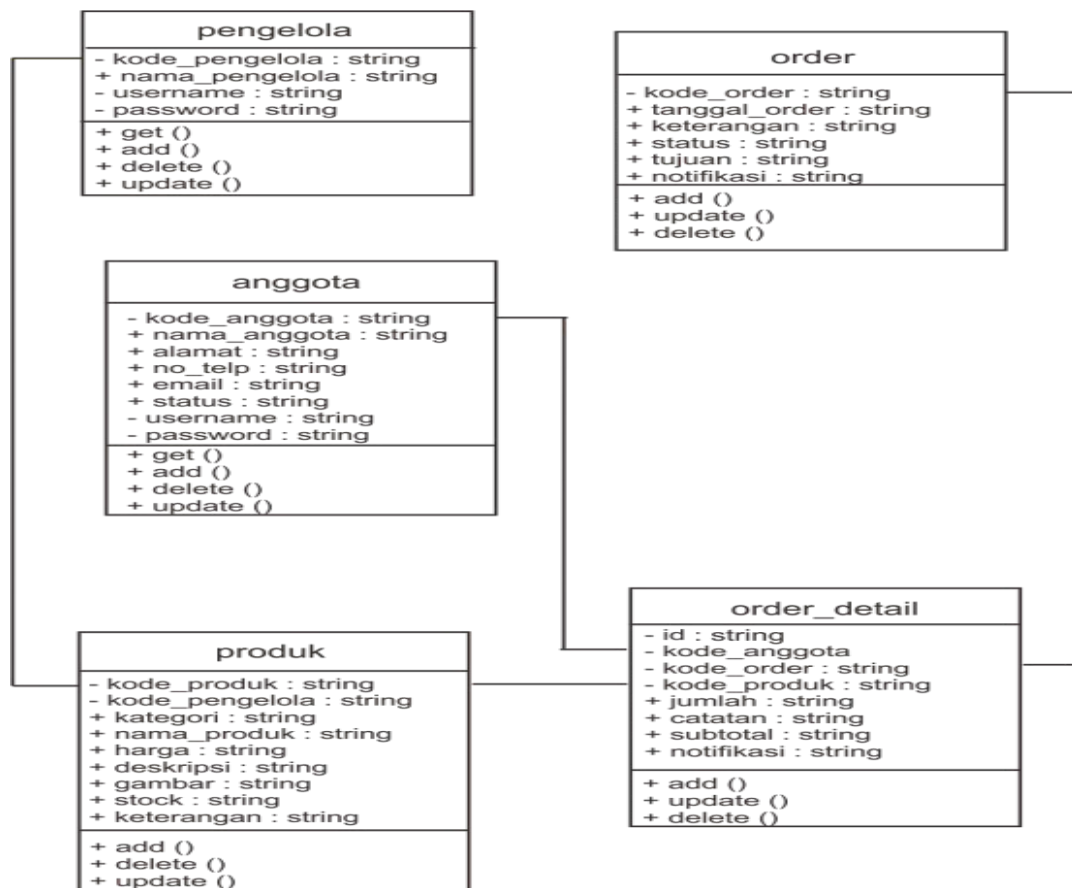
Use case diagram menjelaskan manfaat sistem jika dilihat menurut pandangan orang yang berada diluar sistem (*actor*). Diagram ini menunjukkan fungsionalitas suatu sistem atau *class* dan bagaimana sistem berinteraksi dengan dunia luar. *Use case diagram* aplikasi *E-Marketplace Catering* dijelaskan pada gambar 4.11.



Gambar 4.11 Use Case Diagram *E-Marketplace Catering*.

4.3.3 Class Diagram.

Class Diagram memperlihatkan hubungan antar *class* dan penjelasan detail tiap-tiap *class* di dalam model desain dari suatu sistem. Selama proses analisis, *class diagram* memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem. Selama tahap desain, *class diagram* berperan dalam menangkap struktur dari semua *class* yang membentuk arsitektur sistem yang dibuat. Berdasarkan *class* yang ada dan fungsinya masing – masing. *Class diagram E-Marketplace Catering* akan ditampilkan di gambar 4.12.



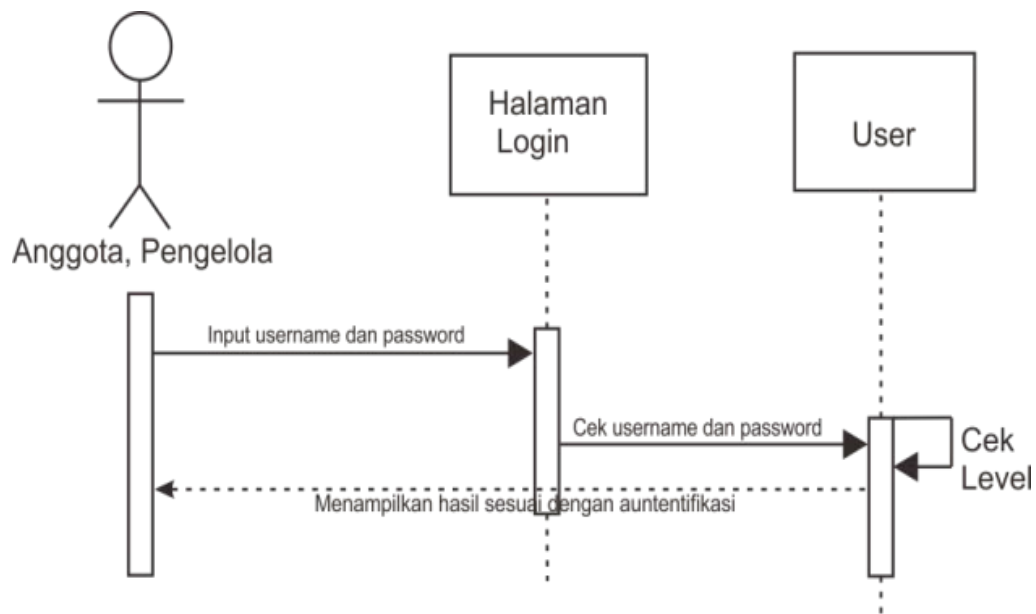
Gambar 4.12 *Class Diagram E-Marketplace Catering*.

4.3.4 Sequence Diagram.

Sequence Diagram adalah suatu diagram yang menggambarkan interaksi antar obyek dan mengindikasikan komunikasi antara obyek-obyek tersebut. Diagram ini juga menunjukkan serangkaian pesan yang diperlukan obyek-obyek yang melakukan suatu tugas atau aksi tertentu.

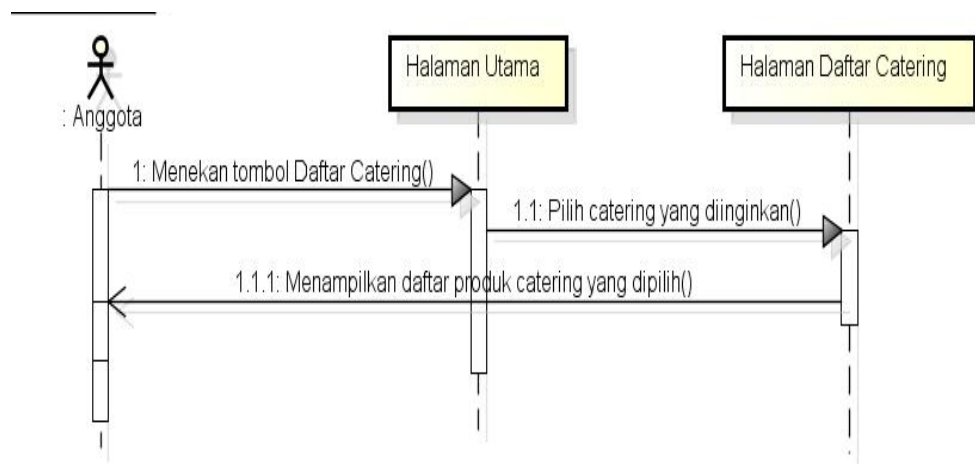
Dari diagram *use case* sebelumnya maka dapat dibuatkan diagram – diagram *sequence* sebagai berikut :

- a. *Sequence* diagram untuk *login* menggambarkan perilaku *user* untuk masuk kedalam sistem. Diagramnya akan dijelaskan pada gambar 4.13.



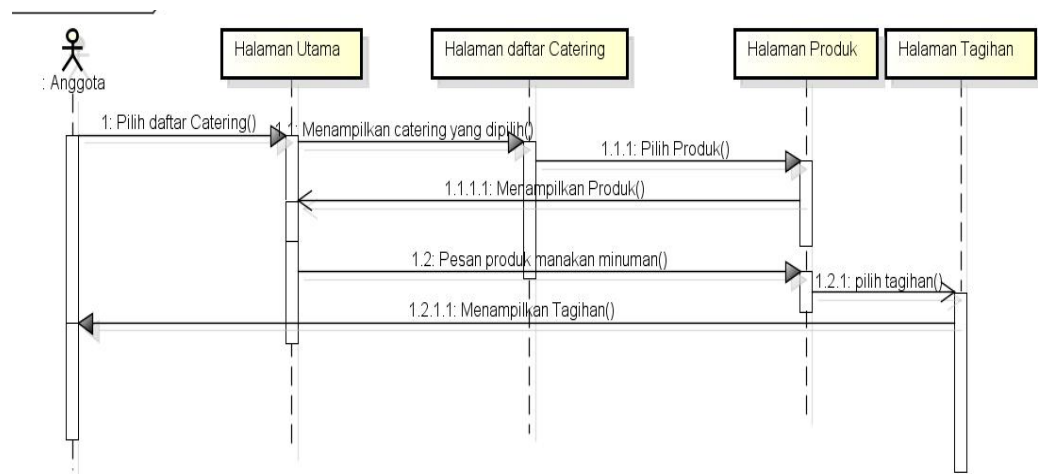
Gambar 4.13 *Sequence* Diagram *login*.

- b. *Sequence* Diagram untuk daftar *E-Marketplace Catering* menggambarkan perilaku *user* dalam melihat daftar menu makanan dan minuman yang telah tercantum sesuai masing-masing menu makanan yang ada pada daftar pengelola. Diagramnya dijelaskan pada gambar 4.14.



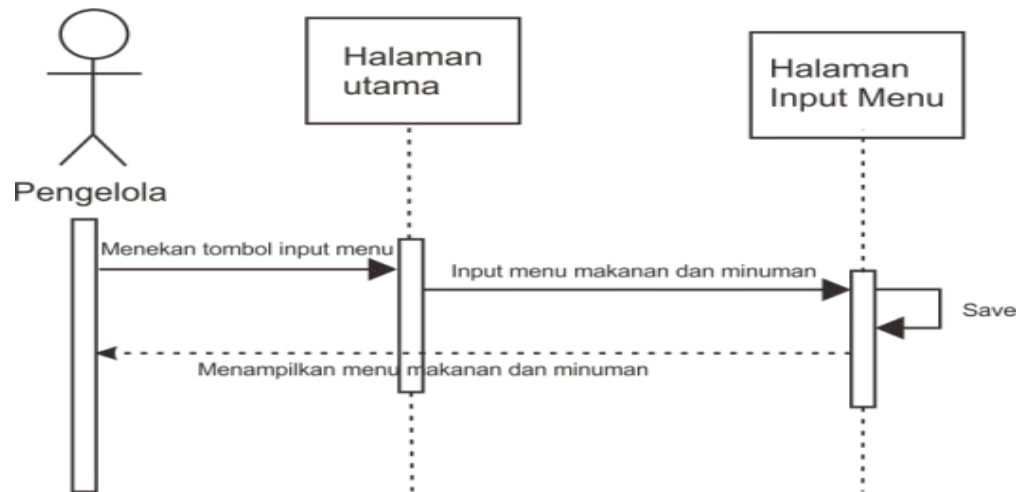
Gambar 4.14 Sequence Diagram daftar *E-Marketplace Catering*.

- c. *Sequence Diagram order* menggambarkan perilaku *user* dalam menentukan menu makanan dan minuman yang akan dipesan. Diagramnya dijelaskan pada gambar 4.15.



Gambar 4.15 Sequence Diagram *order*.

- d. *Sequence* Diagram produk menggambarkan perilaku pengelola dalam meng-*input* menu makanan dan minuman. Diagramnya dijelaskan pada gambar 4.16.



Gambar 4.16 *Sequence* Diagram input produk.

4.4 Tujuan Perancangan Program Aplikasi

Permasalahan yang ada saat ini adalah para pengusaha catering belum mempunyai wadah untuk mempromosikan produk dan pemesanan online dalam bentuk *E-marketplace*. Disisi lain konsumen juga mengalami kesulitan dalam mencari informasi tentang catering yang terbaik. Solusi yang di tawarkan yaitu dengan membuat Sistem Informasi Aplikasi “*E-Marketplace Catering*” dirancang sebagai media alternatif dalam melakukan pemesanan produk makanan dan minuman untuk masyarakat agar mudah dalam menyeleksi catering dengan efektif dan efisien tanpa perlu keluar rumah. Masyarakat yang ingin memesan menu Produk catering berupa makanan dan minuman dapat mengakses sistem ini menggunakan *handphone* berbasis *android* sebagai *mobile device*-nya.

BAB 5

PERANCANGAN PROGRAM APLIKASI

Pada bab ini akan dibahas mengenai elemen dalam perancangan program aplikasi “*E-Marketplace Catering*” yaitu spesifikasi bentuk dokumen masukan, spesifikasi bentuk dokumen keluaran, normalisasi, spesifikasi file dan rancangan tampilan layar.

5.1 Spesifikasi Bentuk Dokumen Masukan

Dalam sub-bab ini akan dijelaskan tentang bentuk dokumen masukan (*input*) yang dilakukan kepada aplikasi. Bentuk dokumen akan dijelaskan dalam bentuk *softcopy* karena aplikasi “*E-Marketplace Catering*” tidak menggunakan *form* fisik dalam dokumen-dokumennya. Aplikasi “*E-Marketplace Catering*” memiliki dua *form input* pada *interface* Isi order dan *Input* produk makanan dan minuman, rincian lebih lanjut dijelaskan sebagai berikut

Nama Dokumen	: Formulir order
Fungsi	: Untuk mengisi pesanan
Frekuensi	: Setiap ada pemesanan
Jumlah	: Satu halaman (<i>activity</i>)

NAMA CATERING

Kode Produk:	PRD001
Nama Produk:	Nama Makanan
Kategori:	Makanan
Harga:	Rp...
Deskripsi:	Enak
Keterangan:	-
Jumlah:	1
Pesan:	-

Kembali

Order Now

Gambar 5.1 *Form* isi order.

Nama Dokumen : Formulir *input* produk
 Fungsi : Untuk meng-*input* menu
 Frekuensi : Setiap meng-*input* produk
 Jumlah : Satu halaman (*activity*)

Gambar 5.2 Form *input* produk

5.2 Spesifikasi Bentuk Dokumen Keluaran

Dalam sub-bab ini akan dijelaskan tentang bentuk dokumen keluaran (*output*) yang dilakukan kepada aplikasi. Bentuk dokumen dijelaskan dalam bentuk *softcopy*. Aplikasi “*E-Marketplace Catering*” memiliki dua *form output* pada *interface* tagihan dan daftar produk, rincian lebih lanjut dijelaskan sebagai berikut :

Nama Dokumen : Tagihan
 Fungsi : menampilkan tagihan menu makanan dan minuman yang telah diorder
 Frekuensi : Setiap akan mengkonfirmasi makanan dan minuman yang telah dipesan
 Jumlah : satu halaman (*activity*)

HALAMAN TAGIHAN

Total Tagihan:

Total Item:

Tujuan Pengiriman:

Gambar 5.3 Bentuk dokumen keluaran tagihan

- Nama Dokumen : Daftar produk
- Fungsi : menampilkan produk makanan dan minuman
- Frekuensi : Setiap pengelola telah meng-*input* produk yang tersedia pada *E-Marketplace Catering*.
- Jumlah : satu halaman (*activity*)

Daftar Produk

Nama makanan atau minuman
kategori, kode produk,
deskripsi.

Nama makanan atau minuman
kategori, kode produk,
deskripsi.

Nama makanan atau minuman
kategori, kode produk,
deskripsi.

Nama makanan atau minuman
kategori, kode produk,
deskripsi.

Gambar 5.4 Bentuk dokumen keluaran produk

5.3 Normalisasi File

Menurut Jogiyanto (2005:403) normalisasi adalah proses untuk mengorganisasikan file untuk menghilangkan grup elemen yang berulang-ulang. Konsep dan teknik normalisasi ini dikenalkan oleh Dr.E.F Codd di papernya pada tahun 1970 dan 1972. Dalam papernya, E.F. Codd mendefinisikan struktur data yang baru yaitu disebut dengan struktur data hubungan (*relational data structure*). Istilah data hubungan menunjukkan suatu struktur data yang mempunyai hubungan dengan elemen-elemen data lainnya, baik dalam satu atau dalam file lainnya.

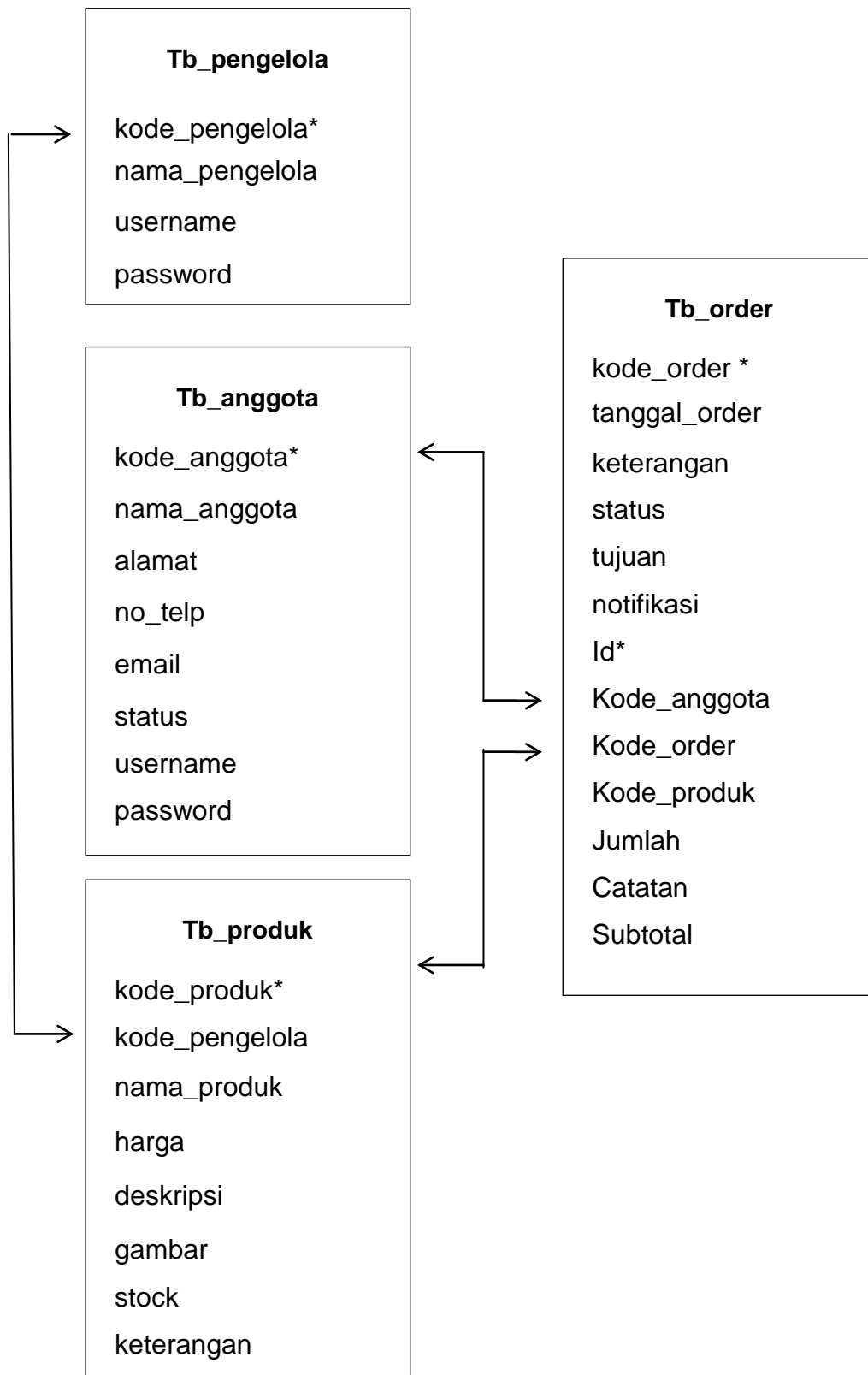
Database dengan struktur data hubungan dapat digambarkan dalam bentuk tabel dua dimensi. Kolom dari tabel menunjukkan atribut dari file. Atribut ini menunjukkan item *data* atau *field*. Kumpulan dari nilai *field* disebut domain. Masing-masing baris dari *record* didalam tabel disebut dengan istilah *tuple*. Tiap-tiap *record* dapat mempunyai satu kunci yang unik dimana *record* ini dapat diidentifikasi. *Field* yang menjadi kunci tersebut disebut dengan *field* kunci (*key field*). Tahapan normalisasi dilakukan hingga tahap 3NF dimana akan dijelaskan pada gambar 5.5 sampai dengan gambar 5.8.

kode_anggota
nama_anggota
alamat
no_telp
email
status
username
password
kode_pengelola
nama_pengelola
username
password
kode_produk
kategori
nama_produk
harga
deskripsi
gambar
stock
keterangan
kode_pengelola
Kode_order
Tanggal_order
kode_anggota
keterangan
status
tujuan
notifikasi
id
kode_order
kode_produk
jumlah
catatan
catatan
sub_total
notifikasi

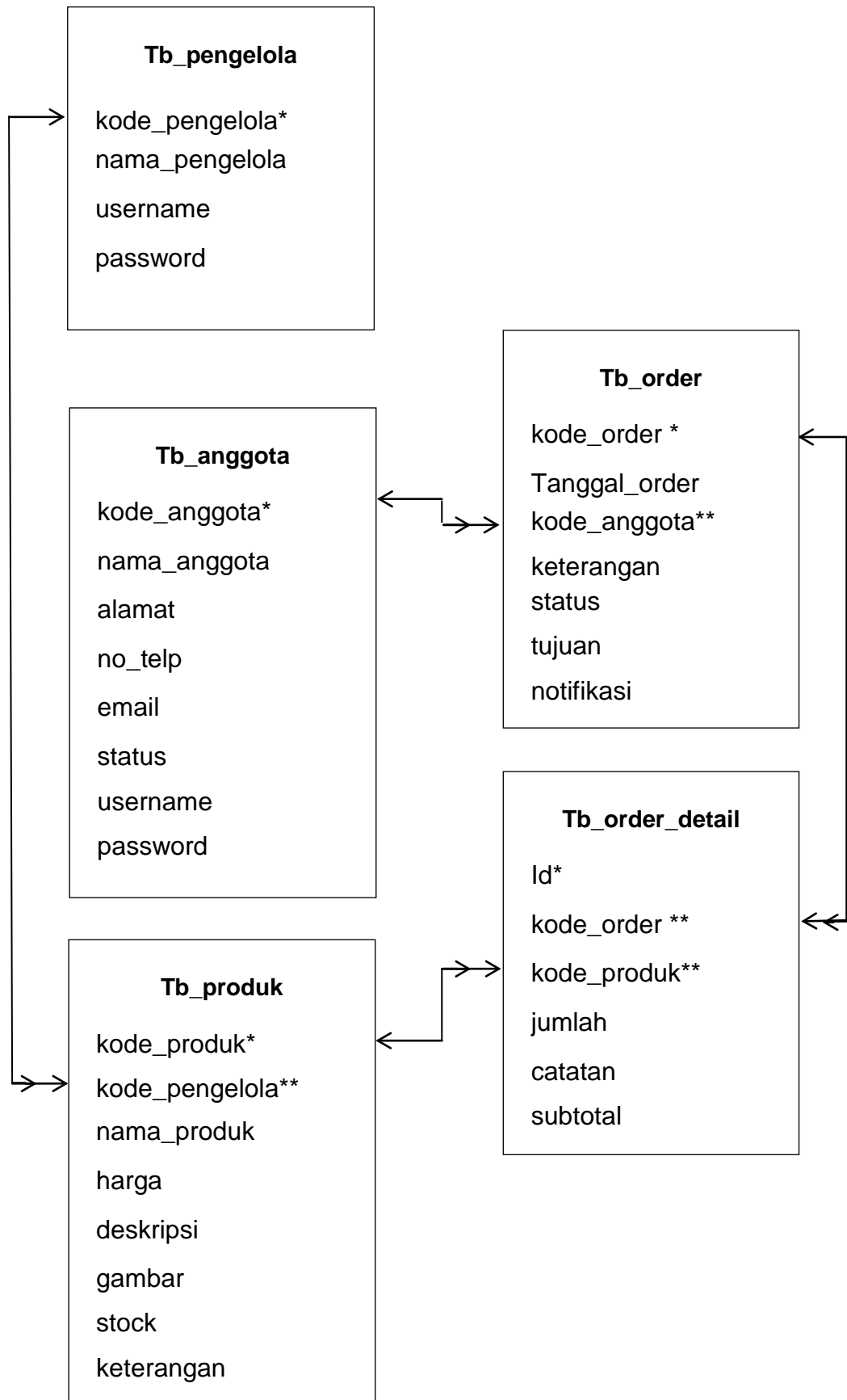
Gambar 5.5 Bentuk tidak normal

kode_anggota*
nama_anggota
alamat
no_telp
email
status
username
password
kode_pengelola*
nama_pengelola
kode_produk*
kategori
nama_produk
harga
deskripsi
gambar
stock
keterangan
kode_order*
tanggal_order
status
tujuan
id*
jumlah
catatan
subtotal

Gambar 5.6 Bentuk normal pertama (1NF)



Gambar 5.7 Bentuk normal kedua (2NF)



Gambar 5.8 Bentuk normal ketiga (3NF)

5.4 Spesifikasi File

Penjelasan tabel-tabel yang digunakan dalam program yang diusulkan serta *field* yang terdapat pada *file database* yang akan dibangun sering disebut dengan spesifikasi file. Tabel-tabel tersebut akan menampung data Informasi yang digunakan dalam aplikasi. Spesifikasi file yang digunakan untuk aplikasi “*E-Marketplace Catering*” ada lima, yaitu sebagai berikut :

1. Tabel / file Pengelola

Nama database : *db_E-Marketplace Catering*
 Nama tabel / file : *pengelola*
 Media file : *hosting*
 Primary key : *kode pengelola*
 Panjang record : *45 characters*

Tabel 5.1 Spesifikasi file / tabel “pengelola”

No.	Nama Field	Tipe data	Panjang field	Ket.
1.	kode_pengelola	varchar	10	Key
2.	nama_pengelola	varchar	15	
3.	username	varchar	10	
4.	password	varchar	10	

2. Tabel anggota

Nama database : *db_E-Marketplace Catering*
 Nama tabel / file : *anggota*
 Media file : *hosting*
 Primary key : *kode anggota*
 Panjang record : *125 characters*

Tabel 5.2 Spesifikasi file / tabel “anggota”

No.	Nama Field	Tipe data	Panjang field	Ket.
1.	kode_anggota	varchar	10	Key
2.	nama_anggota	varchar	15	
3.	alamat	varchar	30	
4.	no_telp	varchar	15	
5.	email	varchar	20	
6.	status	varchar	10	

7.	username	varchar	15	
8.	password	varchar	10	

3. Tabel produk

Nama database : *db_E-Marketplace Catering*

Nama tabel / file : produk

Media file : *hosting*

Primary key : kode_produk

Panjang record : 185 characters

Tabel 5.3 Spesifikasi file / tabel "produk"

No.	Nama Field	Tipe data	Panjang field	Ket.
1.	kode_produk	varchar	10	Key
2.	kode_pengelola	varchar	15	
3.	kategori	varchar	20	
4.	nama_produk	varchar	20	
5.	harga	varchar	10	
6.	deskripsi	varchar	50	
7.	gambar	varchar	20	
8.	stock	varchar	20	
9.	keterangan	varchar	20	

4. Tabel order

Nama database : *db_E-Marketplace Catering*

Nama tabel / file : order

Media file : *hosting*

Primary key : kode_order

Panjang record : 71 characters

Tabel 5.4 Spesifikasi file / tabel "order"

No.	Nama Field	Tipe data	Panjang field	Ket.
1.	kode_order	varchar	10	key
2.	tanggal_order	date	10	
3.	keterangan	text	20	
4.	status	varchar	10	
5.	tujuan	text	20	
6.	notifikasi	integer	1	

5. Tabel orde_detail

Nama database : *db_E-Marketplace Catering*
 Nama tabel / file : *order_detail*
 Media file : *hosting*
 Primary key : *id*
 Panjang record : *86 characters*

Tabel 5.5 Spesifikasi file / tabel "order_detail"

No.	Nama Field	Tipe data	Panjang field	Ket.
1.	Id	integer	15	key
2.	kode_anggota	varchar	10	
3.	kode_order	varchar	10	
4.	kode_produk	varchar	10	
5.	jumlah	integer	10	
6.	catatan	varchar	20	
7.	subtotal	integer	10	
8.	notifikasi	integer	1	

5.5 Spesifikasi Program

Dalam sub-bab ini menjelaskan spesifikasi program dari aplikasi "*E-Marketplace Catering*" yang diuraikan dalam penjelasan sebagai berikut :

Spesifikasi program aplikasi "*E-Marketplace Catering*" :

Nama program : *E-Marketplace Catering Online.apk*

Bahasa program : *java, XML*

Fungsi program : Untuk melihat dan memesan menu makanan dan minuman.

Media program : *phone internal storage*

Logika program : *Login* untuk setiap anggota, anggota dapat melihat menu makanan dan minuman, melihat daftar *E-Marketplace Catering* dan mengisi order. Pengelola dapat melihat data anggota yang telah melakukan order makanan atau minuman, memproses dan mengirim makanannya kepada pemesan. Masing -

masing anggota dan pengelola dapat mengganti profil pribadinya.

5.6 Rancangan Tampilan Layar

Perancangan Tampilan Layar (*Logik User Interface*) merupakan tahapan yang harus dikerjakan pada saat proses pembuatan program. Tujuan dari pembuatan rancangan antarmuka pengguna ini adalah agar pengguna dapat mengetahui gambaran umum program yang akan dibuat, sehingga pengguna dapat dengan mudah mengoperasikan program tersebut. Selain itu, perancangan antarmuka pengguna ini juga bertujuan untuk memudahkan dalam pembuatan program secara keseluruhan.

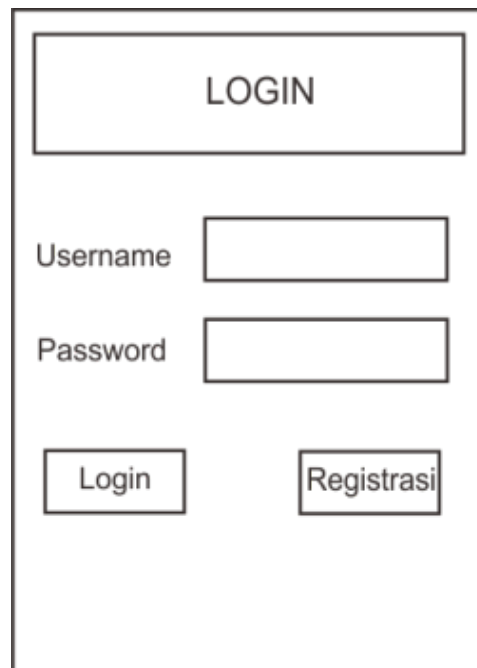
Perancangan *Logik User Interface* dapat dibuat dengan menggunakan *storyboard*. *Storyboard* merupakan gambaran dari program yang akan dibuat. Di dalam sebuah *storyboard* terdapat perancangan dan penggambaran elemen-elemen yang akan terlihat pada layar monitor. Elemen-elemen tersebut meliputi teks, gambar, tata letak, dan bentuk dari tombol-tombol. Pembuatan *storyboard* dilakukan agar terciptanya tampilan secara berurutan pada aplikasi. Pada "*E-Marketplace Catering*" terdapat satu tampilan, yaitu tampilan anggota.

5.6.1 Perancangan Tampilan Anggota

Perancangan tampilan anggota terdapat tujuh tampilan yaitu perancangan halaman *login*, halaman utama, halaman *search*, profil, halaman daftar *E-Marketplace Catering*, halaman *order*, halaman *order detail*, dan halaman *about*.

1) Perancangan halaman *login*

Pada halaman ini adalah tahapan pertama yang harus dilakukan pengguna untuk masuk kedalam sistem. Disini pengguna harus memasukkan *username* dan *password* untuk *login*. Hasil rancangan yang akan diimplementasikan bisa dilihat di gambar 5.9.

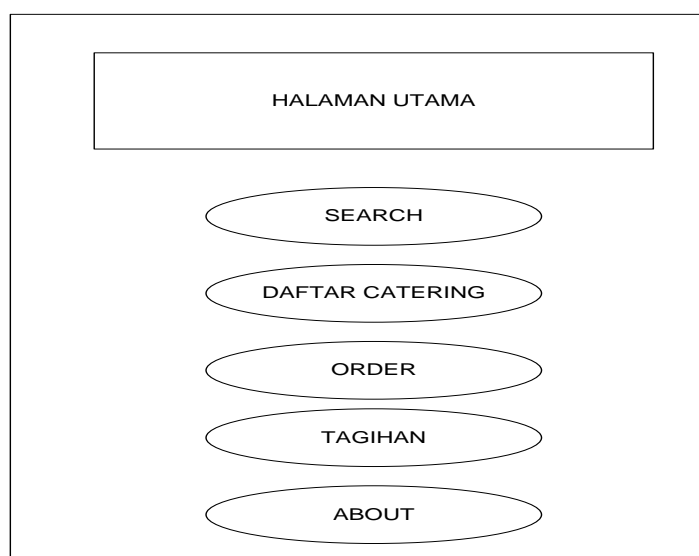


A wireframe for a member login page. At the top, there is a rectangular box containing the word "LOGIN". Below this, the text "Username" is followed by a rectangular input field. Underneath, the text "Password" is followed by another rectangular input field. At the bottom of the form, there are two rectangular buttons: "Login" on the left and "Registrasi" on the right.

Gambar 5.9 Rancangan tampilan *login* anggota

2) Perancangan halaman utama anggota

Dalam halaman utama akan memuat lima menu utama, yaitu *search*, *profil*, *daftar E-Marketplace Catering*, *order*, *about*, *tagihan*. Hasil rancangan yang akan diimplementasikan bisa dilihat di gambar 5.10

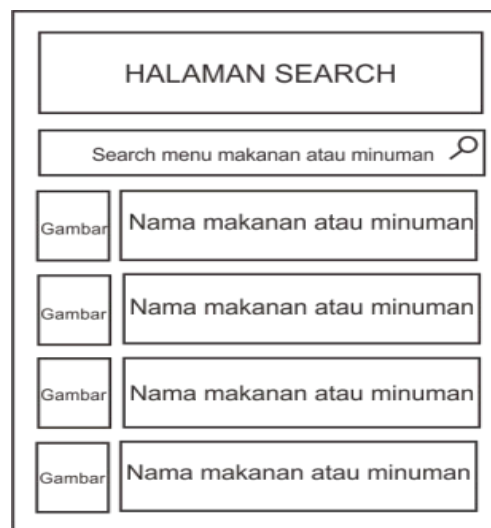


A wireframe for the main member page. At the top, there is a rectangular box containing the text "HALAMAN UTAMA". Below this, five oval-shaped buttons are arranged vertically, containing the text "SEARCH", "DAFTAR CATERING", "ORDER", "TAGIHAN", and "ABOUT" from top to bottom.

Gambar 5.10 Rancangan tampilan menu utama anggota

3) Perancangan halaman menu *search*

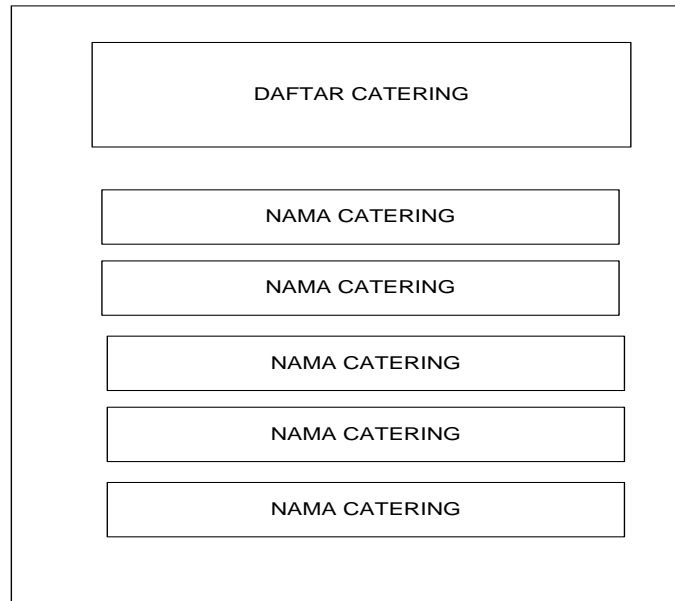
Pada menu ini pengguna dapat melihat tombol *search*. Dengan melakukan *tap* pada tab *search*, pengguna dapat mengetik nama makanan atau minuman yang diinginkan saat itu, apabila menu tersebut ada, maka nama menu makanan atau minuman tersebut akan muncul. Hasil rancangan yang akan diimplementasikan bisa dilihat di gambar 5.11



Gambar 5.11 Rancangan tampilan *search*

4) Perancangan halaman daftar *E-Marketplace Catering*

Pada menu ini pengguna dapat melihat daftar *E-Marketplace Catering*. Dengan melakukan *tap* pada tab daftar *E-Marketplace Catering* pengguna dapat melihat daftar semua *E-Marketplace Catering*. Hasil rancangan yang akan diimplementasikan bisa dilihat di gambar 5.12



Gambar 5.12 Rancangan halaman daftar *E-Marketplace Catering*

5) Perancangan halaman daftar produk

Pada menu ini pengguna dapat melihat daftar produk makanan dan minuman yang tersedia pada masing - masing *E-Marketplace Catering*. Dengan melakukan *tap* pada tab salah satu *E-Marketplace Catering* pengguna dapat melihat daftar semua produk makanan dan minuman yang telah disediakan oleh *E-Marketplace Catering*. Hasil rancangan yang akan diimplementasikan bisa dilihat di gambar 5.13

Daftar Produk
Nama makanan atau minuman kategori, kode produk, deskripsi.
Nama makanan atau minuman kategori, kode produk, deskripsi.
Nama makanan atau minuman kategori, kode produk, deskripsi.
Nama makanan atau minuman kategori, kode produk, deskripsi.

Gambar 5.13 Rancangan halaman daftar produk

6) Perancangan halaman *order*

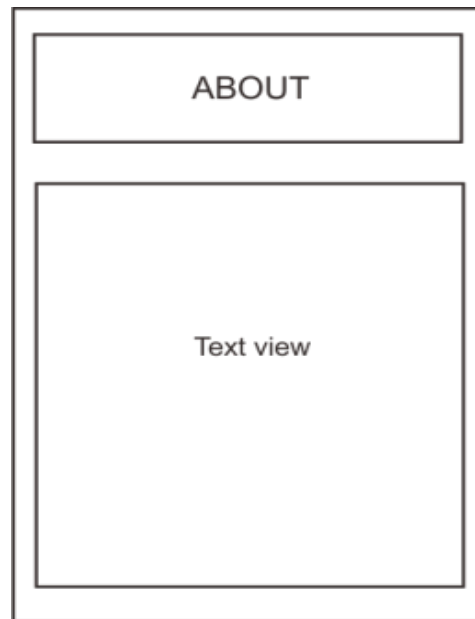
Saat pengguna berada dalam halaman *order* dan melakukan pemesanan makanan dan minuman pada *handphone* maka pengguna akan mendapati tampilan *order*. Hasil rancangan yang akan diimplementasikan bisa dilihat di gambar 5.14

<div style="border: 1px solid black; width: 50px; height: 20px; margin: 0 auto; display: flex; align-items: center; justify-content: center;">Gambar</div>
Kode produk : Nama produk: Kategori Harga Deskripsi: Keterangan: Status: Jumlah: Pesan:
<div style="display: flex; justify-content: space-around; margin-bottom: 5px;"> <div style="border: 1px solid black; padding: 2px 10px;">Ubah</div> <div style="border: 1px solid black; padding: 2px 10px;">Hapus</div> </div> <div style="border: 1px solid black; padding: 2px 20px; text-align: center;">Kembali</div>

Gambar 5.14 Rancangan tampilan halaman order anggota

7) Perancangan halaman *about*

Halaman ini dapat menampilkan informasi mengenai aplikasi *E-Marketplace Catering*. Setelah pengguna melakukan *tap* pada tombol *about* maka akan muncul tampilan rancangan yang akan diimplementasikan bisa dilihat di gambar 5.15



Gambar 5.15 Rancangan tampilan *about*

5.6.2 Perancangan Tampilan Pengelola

Perancangan tampilan pengelola terdapat delapan tampilan yaitu perancangan halaman *login*, halaman utama, halaman profil, halaman anggota, halaman produk, halaman order, halaman arsip.

1) Perancangan halaman *login*

Pada halaman ini adalah tahapan pertama yang harus dilakukan pengguna untuk masuk kedalam sistem. Disini pengguna harus memasukkan *username* dan *password* untuk *login*. Hasil rancangan yang akan diimplementasikan bisa dilihat di gambar 5.16

Header

Login Pengelola

Username:

Password :

Login Reset

FOOTER

Gambar 5.16 Rancangan tampilan *login* pengelola

2) Perancangan halaman utama pengelola

Dalam halaman utama akan memuat tujuh menu utama, yaitu *home*, *profil*, *anggota*, *produk*, *order*, *arsip*, dan *logout*. Hasil rancangan yang akan diimplementasikan bisa dilihat di gambar 5.17

Home Profil Anggota Produk Order Arsip Logout

Halaman Utama

FOOTER

Gambar 5.17 Rancangan tampilan halaman utama pengelola

3) Perancangan halaman profil pengelola

Pada menu ini pengguna dapat melihat tombol profil. Dengan melakukan *tap* pada tab profil, Saat pengguna melakukan aksi *tap* ke tombol “Profil” maka pengguna akan mendapati tampilan yang berisi kode pengelola, nama pengelola, *username* dan *password* yang dapat diubah sesuai keinginan pengelola. Hasil rancangan yang akan diimplementasikan bisa dilihat di gambar 5.18

Gambar 5.18 Rancangan tampilan halaman profil pengelola

4) Perancangan halaman anggota

Pada menu ini pengguna dapat melihat daftar anggota. Dengan melakukan *tap* pada menu anggota, pengguna dapat melihat daftar semua anggota. Hasil rancangan yang akan diimplementasikan bisa dilihat di gambar 5.19

Gambar 5.19 Rancangan tampilan halaman anggota

5) Perancangan halaman produk

Pada menu ini pengguna dapat melihat daftar produk makanan dan minuman dan dapat menambah serta mengurangi apabila *stock* telah habis. Hasil rancangan yang akan diimplementasikan bisa dilihat di gambar 5.20

Gambar 5.20 Rancangan tampilan halaman produk

6) Perancangan halaman *order* pengelola

Saat pengguna berada dalam halaman *order*, pengelola dapat melihat *notifikasi* order dari anggota yang telah melakukan pemesanan makanan. Hasil rancangan yang akan diimplementasikan bisa dilihat di gambar 5.21

No	Nama Produk	Kategori	Harga	Sub Total	Catatan	Gambar	Approve

Gambar 5.21 Rancangan tampilan halaman *order* pengelola

7) Perancangan halaman arsip

Pada menu ini pengguna dapat melihat tombol arsip. Dengan melakukan *tap* pada menu arsip, maka pengguna akan mendapati tampilan yang berisi daftar *order* makanan yang telah di *approve* sebelumnya. Hasil rancangan yang akan diimplementasikan bisa dilihat di gambar 5.22

Home	Profil	Anggota	Produk	Order	Arsip	Logout
------	--------	---------	--------	-------	-------	--------

Halaman Order						
Menu Order						
Kode Order :						
Tanggal :						
Anggota :						
Status :						
Keterangan :						
Tujuan :						

No	Nama Produk	Kategori	Harga	Sub Total	Catatan	Gambar

Total Tagihan :

Gambar 5.22 Rancangan tampilan halaman arsip

BAB 6

IMPLEMENTASI PROGRAM APLIKASI

Pada bab ini akan dibahas mengenai tahap akhir perancangan aplikasi yaitu implementasi, dari aplikasi “*E-Market Place Catering*” yang telah dibuat. Penjelasan implementasi meliputi spesifikasi *software* dan *hardware*, rancangan fisik *user interface*, dan peralihan program aplikasi,

6.1 Spesifikasi *Software* dan *Hardware*

Dalam sub-bab ini akan dijelaskan mengenai perangkat lunak (*software*) dan perangkat keras (*hardware*) yang digunakan dalam perancangan dan implementasi aplikasi “*E-Market Place Catering*”. Perangkat keras yang digunakan oleh penulis adalah sebuah komputer laptop dengan spesifikasi sebagai berikut :

- 1) *CPU* : AMD C-60 APU With Radeon (tm) HD Graphics (2 CPUs) 1,0GHz
- 2) *Memory RAM* : 2 GB
- 3) *Kapasistas Hardisk* : 500 GB *Harddisk Drive*
- 4) *Monitor* : 14.”

dan, sebuah perangkat *smartphone android* sebagai alat ujicoba dengan spesifikasi sebagai berikut :

- 1) *Processor* : quad-core 1,2GHz ARM Cortex-A7 32-bit
- 2) *Sistem Operasi* : Google Android 4.4 KitKat + IDOS
- 3) *Layar* : 4 Inchi WVGA 480 x 800 piksel, TFT Kapasitif, ~233ppi
- 4) *Memory RAM* : 1 GB
- 5) *Memory Internal*:8 GB

Perangkat lunak yang digunakan meliputi sistem operasi laptop dan *smartphone* serta *integrated development environment (compiler)* dengan spesifikasi sebagai berikut :

- 1) Sistem operasi Microsoft Windows 7

- 2) *Xampp 1.8.0, Apache/2.4.2 (Win32) OpenSSL/1.0.1c PHP/5.4.4*
- 3) *Android OS 4.4.2 (Kitkat)*
- 4) *Web Hosting www.000webhost.com*

6.2 Rancangan Fisik *User Interface*

Dalam sub-bab ini akan digambarkan bagaimana tampilan *user interface* aplikasi “E-Market Place Catering” setelah selesai dibuat. Rancangan *interface* Berikut adalah tampilan dari aplikasi “E-Market Place Catering”.

6.2.1 Tampilan Halaman Default

Ketika aplikasi dijalankan maka akan tampil halaman default berisi Tampilan default halaman utama yang berisi tampilan produk terlaris akan terlihat di gambar 6.1.



Gambar 6.1 Tampilan Halaman *Default*.

6.2.2 Tampilan Halaman Registrasi Anggota

Sebelum anggota bisa *login* Untuk bisa masuk dan memesan produk maka anggota harus melakukan registrasi terlebih dahulu dengan memilih *menu registrasi* di halaman depan Tampilan halaman *registrasi* akan terlihat di gambar 6.2.

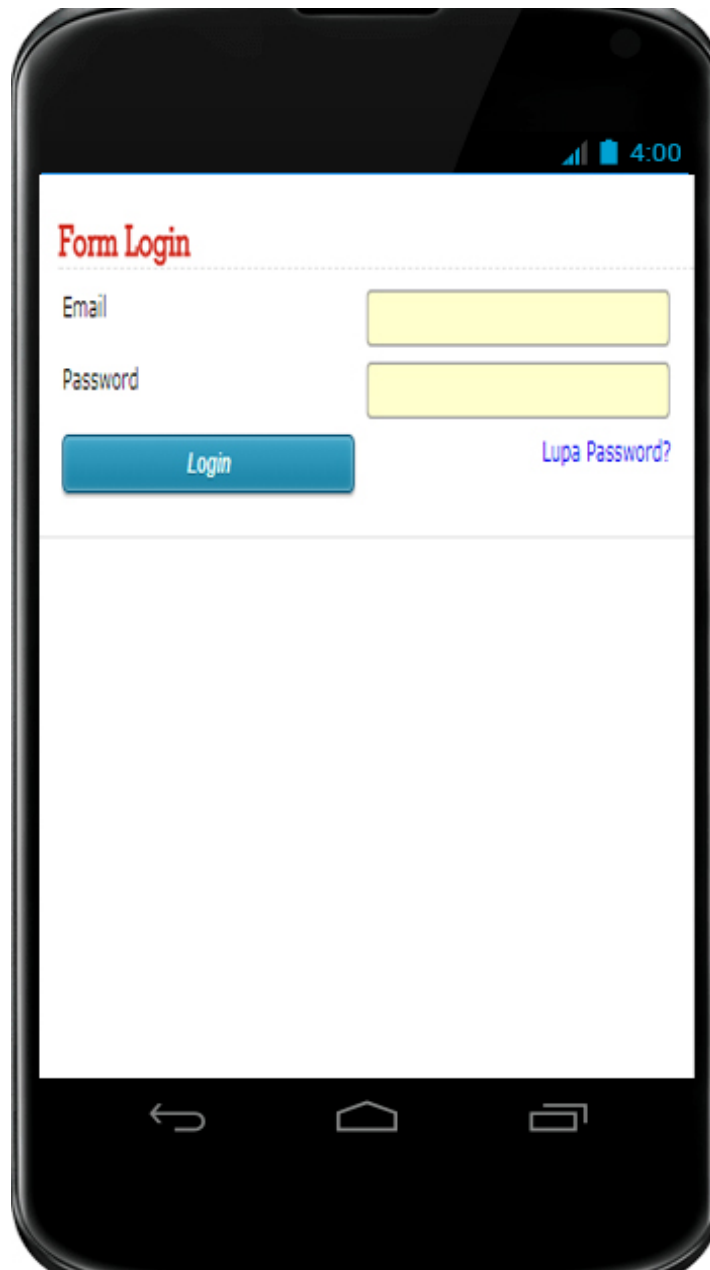


The image shows a mobile application interface for a registration form. The form is titled "Form Register" in red text. It contains several input fields: "Nama Lengkap", "Password", "Alamat Pengiriman" (a large text area), "Nomor Rekening", "Email", and "Kota Tujuan" (a dropdown menu). Below the "Kota Tujuan" field, there are two lines of explanatory text: "*) Apabila tidak terdapat nama kota tujuan Anda, pilih" and "**) Ongkos kirim dihitung berdasarkan kota tujuan". A blue verification code "5e530e" is displayed in a grid pattern, with the instruction "(Masukkan 6 kode diatas)" below it. A blue "Daftar" button is located at the bottom of the form. The phone's status bar at the top shows the time as 4:00 and signal strength icons. The Android navigation bar is visible at the bottom.

Gambar 6.2 Tampilan Halaman *Registrasi*.

6.2.3 Tampilan *Login User*

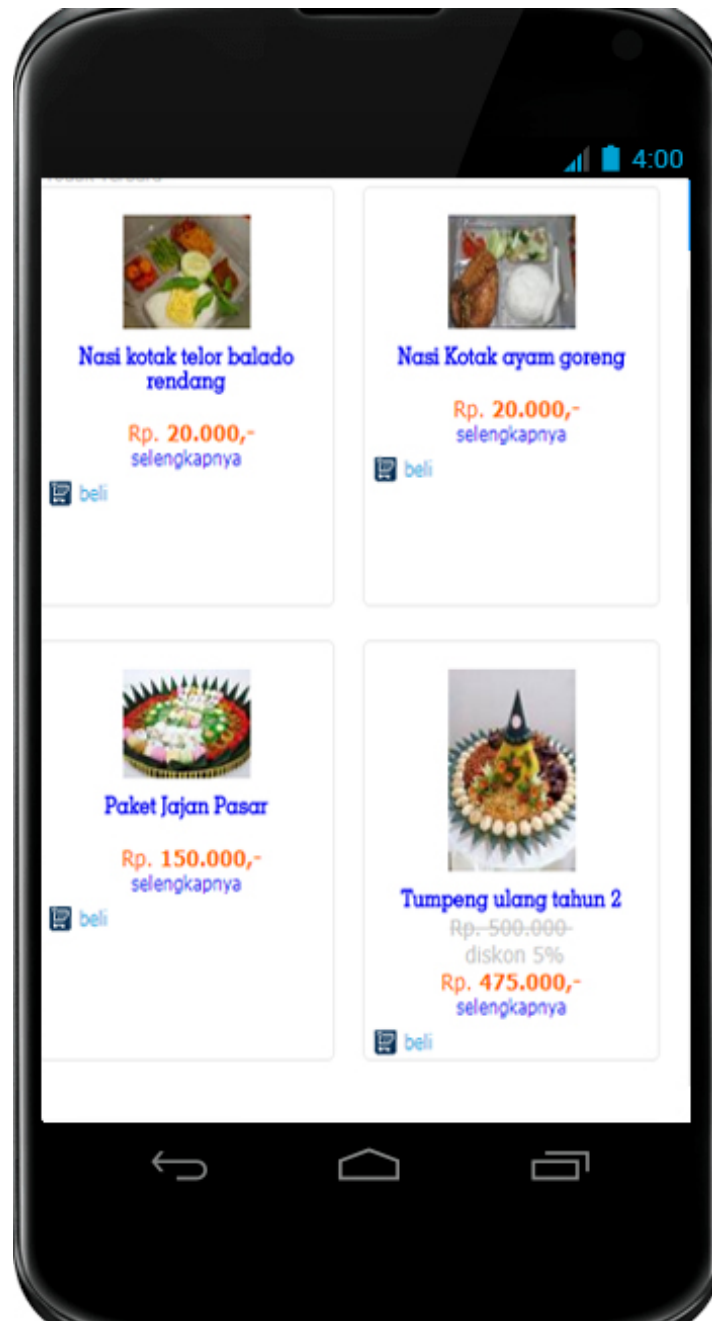
Setelah Berhasil registrasi anggota bisa masuk ke sistem untuk melakukan pemesanan produk. Untuk *login user* klik menu *login* pada halaman utama Tampilan *Login User* akan terlihat di gambar 6.3



Gambar 6.3 Tampilan Halaman *Login User*.

6.2.4 Tampilan Daftar Produk

Tampilan daftar produk memuat gambar produk, harga dan penjelasan produk, jika anggota sudah *login* maka bisa klik beli. Tampilan daftar produk akan ditampilkan pada gambar 6.4.



Gambar 6.4 Tampilan Halaman Daftar Produk.

6.2.5 Tampilan Pesan Produk

Tampilan pesan produk berisi tentang daftar produk yang telah dipesan oleh *user*. Tampilan order akan terlihat di gambar 6.5.



Gambar 6.5 Tampilan Halaman *Order*.

6.2.6 Tampilan Detail Order

Tampilan *detail order* berisi tentang daftar produk yang akan dipesan. Tampilan *detail order* akan terlihat di gambar 6.6.



Gambar 6.6 Tampilan Halaman *Detail Order*.

6.2.7 Tampilan Profil Anggota

Pada tampilan profil, *user* dapat mengganti *profil* sesuai keinginan sendiri. *User* dapat mengganti nama, *email*, alamat, nomor rekening, tujuan. Tampilan *profil* akan terlihat di gambar 6.7.

Form Edit Your Profil

Nama Lengkap	<input type="text" value="abid alkhalfi"/>
Alamat Pengiriman	<input type="text" value="jln. Gunung raya no 69, cireundeu ciputat"/> <small>Alamat pengiriman harus di isi lengkap, termasuk kota/kabupaten dan kode posnya.</small>
Nomor Rekening	<input type="text"/>
Kota Tujuan	<input type="text" value="- Pilih Kota -"/>

*) Apabila tidak terdapat nama kota tujuan Anda, pilih **Lainnya**
**) Ongkos kirim dihitung berdasarkan kota tujuan

Gambar 6.7 Tampilan Halaman *Profil*.

6.2.8 Tampilan *Login Admin*

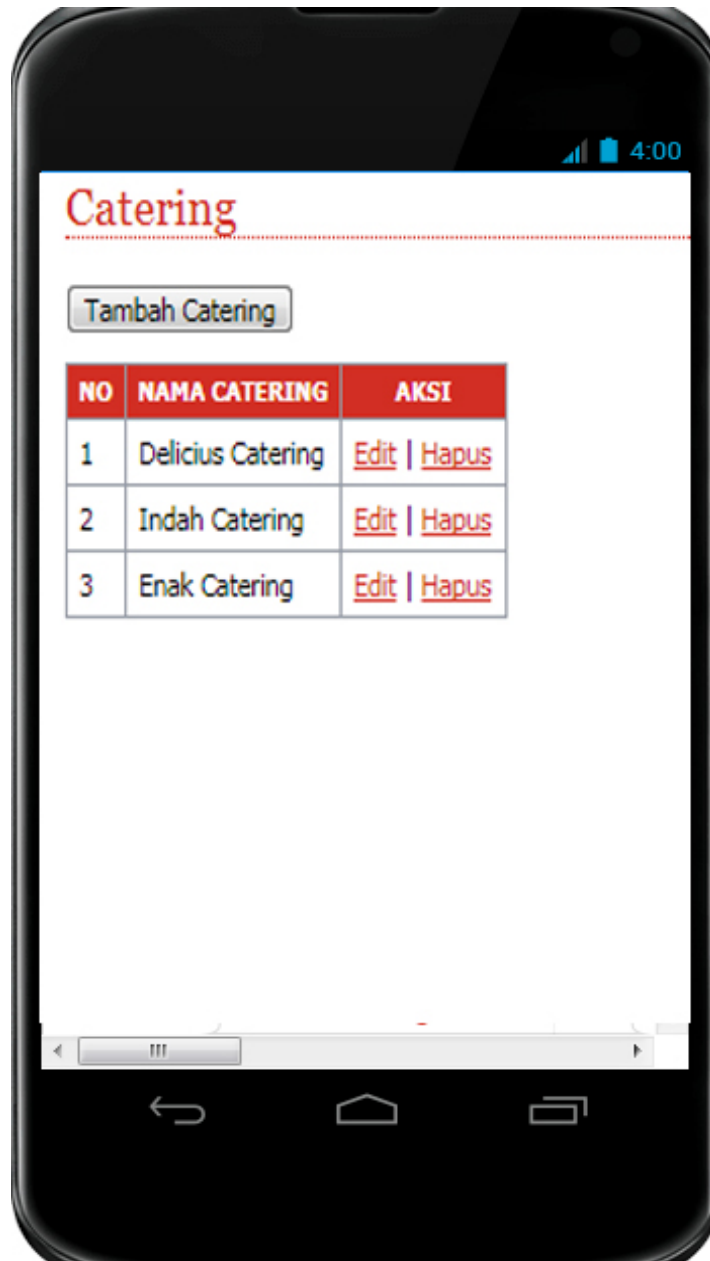
Tampilan *login admin* digunakan *administrator* untuk mengelola data produk, data catering, data order dan laporan. Tampilan *login admin* dapat dilihat pada gambar 6.8.



Gambar 6.8 Tampilan Halaman *Login Admin*.

6.2.9 Tampilan Halaman Daftar *Catering*

Halaman ini dapat menampilkan informasi mengenai daftar catering yang terdaftar. *Admin* bisa menambah data catering, mengedit dan menghapus data catering. Tampilan daftar catering akan terlihat di gambar 6.9.



Gambar 6.9 Tampilan Halaman *Daftar Catering*

6.2.10 Tampilan Daftar Produk

Halaman ini dapat menampilkan informasi mengenai daftar produk yang terdaftar. *Admin* bisa menambah data produk, mengubah dan menghapus data produk. Tampilan daftar produk akan terlihat di gambar 6.10.

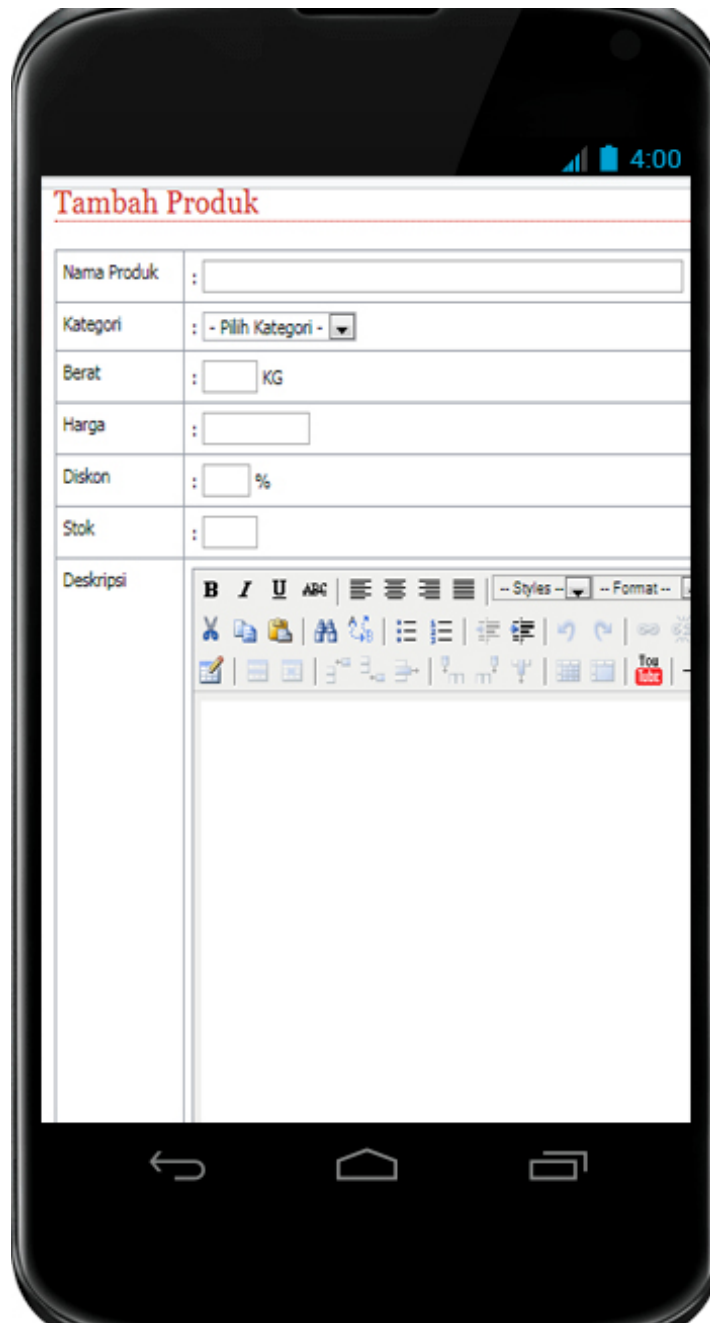


NO	NAMA PRODUK	BERAT(KG)	HARGA
1	Nasi kotak telur balado rendang	0.00	20.000
2	Nasi Kotak ayam goreng	0.00	20.000
3	Paket Jajan pasar komplit	1.00	200.000
4	Paket Jajan Pasar	1.00	150.000
5	Tumpeng ulang tahun 2	4.00	500.000
6	Tumpeng ulang tahun 1	3.00	350.000

Gambar 6.10 Tampilan Halaman Daftar Produk.

6.2.11 Tampilan Halaman Tambah Data Produk

Untuk menambah produk maka dipilih tambah produk selanjutnya input pada *form* yang tersedia kemudian klik simpan. Tampilan halaman tambah produk akan terlihat di gambar 6.11.



The screenshot displays a mobile application interface for adding a product. The title bar at the top reads "Tambah Produk" in red text. The form consists of several input fields: "Nama Produk" (text input), "Kategori" (dropdown menu with "- Pilih Kategori -"), "Berat" (text input followed by "KG"), "Harga" (text input), "Diskon" (text input followed by "%"), and "Stok" (text input). Below these fields is a "Deskripsi" section featuring a rich text editor with a toolbar containing icons for bold, italic, underline, text color, background color, bulleted list, numbered list, link, unlink, and other formatting options. The bottom of the screen shows the standard Android navigation bar with back, home, and recent apps buttons.

Gambar 6.11 Tampilan Halaman Tambah Produk.

6.2.12 Tampilan Halaman Pengguna.

Untuk pengguna dapat mengisi *password*. Klik pengguna kemudian isi *password* sesuai keinginan si pengguna. Tampilan halaman tambah pengguna akan terlihat di gambar 6.12.



Tambah Pengguna	
Username	: Farhan
Password	: Farhan
Nama Lengkap	: Farhan W
Email	: Farhan@gmail.com
Nomor Telepon	: 085718831885
Level	: User
<input type="button" value="Simpan"/> <input type="button" value="Batal"/>	

Gambar 6.12 Tampilan Halaman Tambah Pengguna.

6.2.13 Tampilan Halaman *Order Detail*

Pengelola bisa melihat *detail order* dari anggota yang berisi produk pesanan, jumlah biaya tujuan pengiriman dan data pemesan Tampilan *detail order* akan terlihat di gambar 6.13.

Detail Order

No. Order	: 27
Tgl. & Jam Order	: 12 Juli 2017 & 14:57:59
Status Order	: <input type="button" value="Baru"/> <input type="button" value="Ubah Status"/>

NAMA PRODUK	BERAT(KG)	JUMLAH	HARGA SATUAN	SUB TOTAL
Tumpeng ulang tahun 2	4.00	1	500.000	475.000
Total Rp. :				475.000
Ongkos Kirim Rp. :				13.000/Kg
Total Berat :				4 Kg
Total Ongkos Kirim Rp. :				52.000
Grand Total Rp. :				527.000

DATA KUSTOMER	
Nama Kustomer	: abid alkhalifi
Alamat Pengiriman	: jln. Gunung raya no 69, cireundeu ciputat
Nomor Rekening	: 67601839
Email	: abid@gmail.com

Gambar 6.13 Tampilan Halaman *Detail Order*

6.3 Peralihan Program Aplikasi

Peralihan program aplikasi proses penghentian penggunaan sistem lama dan memulai penggunaan sistem baru disebut *CUT OVER*. Ada empat pendekatan dasar, yaitu percontohan, serentak, bertahap dan paralel. Adapun penjelasan mengenai empat pendekatan tersebut diatas akan dipaparkan dibawah ini, sebagai berikut :

1. Percontohan (pilot), percontohan adalah suatu sistem percobaan yang diterapkan dalam satu subset dari keseluruhan operasi, satu kantor atau daerah tertentu.

2. Serentak (*Immediate*). Pendekatan yang paling sederhana adalah dengan beralih dari sistem lama ke sistem baru pada satu hari tertentu. Namun pendekatan ini hanya layak bagi perusahaan kecil atau sistem kecil karena permasalahan waktu menjadi makin besar saat skala operasi meningkat.
3. Bertahap (*Phased*). Dalam *cut-over* bertahap, sistem baru digunakan bagian perbagian pada suatu waktu.
4. Paralel (*parallel*). *Cut – over paralel* mengharuskan sistem lama dipertahankan sampai sistem baru telah diperiksa secara menyeluruh, pendekatan ini memberikan pengaman yang paling baik terhadap kegagalan tetapi juga paling mahal karena kedua sumber daya harus dipertahankan.

Cut-over menandakan berakhirnya bagian pengembangan dari siklus hidup sistem. Aplikasi “*E-MarketPlace Catering*” ini mengadaptasi dari peralihan aplikasi “Serentak” atau “*immediate*” yaitu peralihan sistem manual ke sistem yang memanfaatkan media yang lebih efisien dan cepat yaitu *internet* dan *gadget*.

BAB 7

PENUTUP

Setelah melakukan penelitian dan perancangan terhadap aplikasi “*E-Marketplace Catering*” maka Pada bab ini dijelaskan mengenai kesimpulan dan saran dari penulis.

7.1 Simpulan

Berdasarkan penelitian dan analisis, perancangan, implementasi, dan uji coba yang telah dilakukan untuk Perancangan dan Implementasi *Marketplace Catering Berbasis Android*, dapat diperoleh kesimpulan yaitu sebagai berikut :

- a. Aplikasi ini membantu untuk mempermudah *user* atau pengguna dalam mencari katering dan melakukan pemesanan produk katering
- b. Aplikasi ini membantu para pengelola katering dalam memasarkan produk mereka secara lebih luas.

7.2 Saran

Untuk pengembangan aplikasi “*E-marketplace Catering*” lebih lanjut agar semakin memberikan manfaat bagi pengguna, ada beberapa hal yang dapat dijadikan bahan kajian lebih lanjut yaitu :

- a. Penambahan fitur *Global Positioning System* (GPS) agar fitur aplikasi menjadi semakin lengkap.
- b. Sistem konfirmasi bisa dikembangkan otomatis melalui email atau SMS jika pesanan sudah di proses atau dikirim.
- c. Sistem pembayaran yang ada saat ini yaitu tranfer ke bank dan pembayaran langsung pada saat produk sampai. Pengembangan selanjutnya dapat dilakukan dengan menambahkan fitur pembayaran secara *online* dengan kartu kredit *atau pay pall* dan sejenisnya.

DAFTAR PUSTAKA

- Akbar, Muhammad. K, I, Satoto. R R, Isnanto. 2014. *Pembuatan Aplikasi Layanan Pesan Antar Makanan Pada Sistem Operasi Android*. Sumber: <http://ejournal.undip.ac.id/index.php>. (Diakses pada tanggal 8 Maret 2016)
- Akbarul, Arif. 2013. *Live Coding! 9 Aplikasi Android Buatan Sendiri*. Penerbit Andi, Yogyakarta.
- Ali Zaki, 2007. *Pengertian Aplikasi*. Andi. Yogyakarta
- Andrews, Sudhir. (2009). *Hotel Front Office: A Training Manual (second edition)*. West Patel Nagar, New Delhi: Tata McGraw-Hill.
- Barbaran, Regina S. & Durocher, Joseph F., *Succesfil Restaurant Design*, Van Nostrand Reinhold, New York, 1992.
- Brunn Peter, Jensen Martin, Skovgaard Jakob. 2002. *e-Marketplaces: Crafting A Winning Strategy*. *European Management Journal* Vol. 20, No. 3, pp. 286–298.
- Connolly, T., Begg, C. 2010. *Database Systems: a practical approach to design, implementation, and management*. 5th Edition. America: Pearson Education
- Dharma Kasman, Akhmad. 2015. *Trik Kolaborasi Android dengan PHP dan MySql*. CV.Lokomedia, Yogyakarta.
- Hengky W. Pramana, (2006), *Aplikasi Inventory Berbasis Access 2003*, Elex Media Komputindo, Jakarta.
- Irawan. 2014. *Aplikasi Android dengan Eclipse*. Maxikom, Palembang.

- Inmon, William H. 2005. "Building The Data Warehouse (4th ed.)". Indianapolis :Wiley Publishing, Inc
- I Wayan, Suartana, 1987: 14. *Prosedur Reservation Untuk Wisatawan Sebagai Media Pemesanan Fasilitas Pariwisata*. Jakarta: Balai Pustaka.
- Jogiyanto. 2005. *Analisis dan Desain Sistem Informasi Pendekatan Terstruktur*. Penerbit Andi, Yogyakarta.
- Kaye Chon & Amy Chan, *The International Conference on Impact of Movies and Television on Tourism*, The Hongkong Polytechnic University, 2009.
- Kardigantara, suseno 2007, Diktat: Operasional Katering : STPB Bandung
- Munawar. (2005), *Pemodelan Visual dengan UML*, Graha Ilmu, Yogyakarta, 17-100.
- Pratama , Antonius Nugraha Widhi. (2010) *Codeigniter: Cara Mudah Membangun Aplikasi PHP*. Jakarta: Mediakita.
- Razak, Abdul. 2014. *Perancangan Sistem Informasi Pemesanan Makanan dan Minuman (e-Menu) Berbasis Platform Android pada Restoran Moca Cafe*. Sumber : https://repository.telkomuniversity.ac.id/pustaka/files/102470/jurnal_eproc/aplikasi-sistem-pemesanan-makanan-dan-minuman-berbasis-android-studi-kasus-kedai-soe-soe-delivery.pdf. (Diakses pada tanggal 5 maret 2016).
- Safaat H, Nazruddin. 2013. *Aplikasi Berbasis Android, Berbagai Implementasi dan Pengembangan Aplikasi Mobile Berbasis Android*. Informatika, Bandung.
- Safaat, Nazruddin H. 2015. *Rancang Bangun Aplikasi Multiplatform*. Informatika, Bandung.

Santoso, Harip (2006). Membuat Database pada SQL Server 2000 Menggunakan VB 6, PT Elexmedia Komputindo : Jakarta.

Sugiyono. 2005. Metode penelitian Administrati. Alfabeta, Bandung

Sutabri, Tata. 2012. *Konsep Sistem Informasi*. Penerbit Andi, Yogyakarta.

Yasin, Verdi 2012. Rekayasa Perangkat Lunak Berorientasi Objek, Mitra Wacana Media. Jakarta

Wicaksono, Yogi (2008). "Membangun Bisnis Online dg Mambo++ CD". PT. Elex Media Komputindo, Jakarta

W, M, Andreas. B, Irawan. A, B, Osmond. 2013. *Aplikasi Sistem Pemesanan Makanan Dan Minuman Berbasis Android (Studi Kasus: Kedai Soe Soe Delivery)*. Sumber: https://repository.telkomuniversity.ac.id/pustaka/files/102470/jurnal_eproc/aplikasi-sistem-pemesanan-makanan-dan-minuman-berbasis-android-studi-kasus-kedai-soe-soe-delivery.pdf. (Diakses pada tanggal 5 maret 2016).

https://id.wikipedia.org/wiki/Jasa_boga

DAFTAR RIWAYAT HIDUP



NPM	: 121100019
Nama	: Muhammad Andrean Eka Saputra
Kota & tanggal lahir	: Jakarta, 17 September 1994
Jenis Kelamin	: Laki-laki
Agama	: Islam
Status	: Belum Menikah
Warga Negara	: Indonesia
Alamat	: Tebet Barat XIII-A NO.21, RT.010/RW.005, Tebet Barat Tebet – Jakarta Selatan
No.Handphone	: 087884477192
Email	: andreanekasaputra@gmail.com
Riwayat Pendidikan	: 2000 – 2006 SD Angkasa 12 Jakarta Timur 2006 – 2009 SMPN 157 Jakarta Timur 2009 – 2012 SMK Utama Bekasi 2012 – Sekarang Fakultas Teknologi Informasi URINDO
Riwayat Kerja	: 2012 – Sekarang Technical Support PT. Megawastu Solusindo

Jakarta, 16 Juli 2017

M. Andrean. E. S
NPM : 121100019