

BAB III

METODOLOGI PENELITIAN

3.1 Alat dan Bahan

Dalam pembuatan aplikasi digunakan alat dan bahan untuk mendukung berjalannya perancangan dan pembuatan aplikasi.

3.1.1 Alat

Alat yang digunakan dalam penelitian ini berupa laptop atau notebook dengan spesifikasi sebagai berikut:

1. Processor : AMD A10-5750M APU with Radeon™
2. Memory : 4096MB RAM
3. Sistem Manufaktur : ASUSTeK COMPUTER INC.
4. Sistem Model : X550DP
5. Sistem Operasi : Windows 10 Pro 64-bit

Dengan menggunakan perangkat lunak yang mendukung, yaitu :

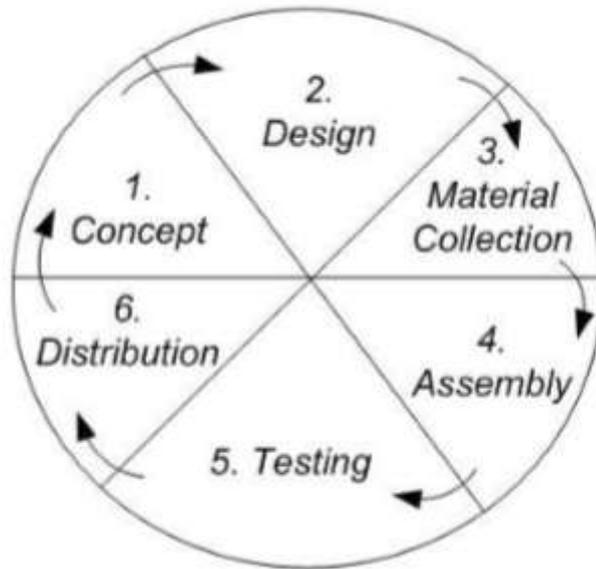
1. NetBeans IDE 8.2
2. Microsoft Office Excel 2016

3.1.2 Bahan

Bahan yang digunakan dalam penelitian ini adalah data-data pemesanan, harga produk, dan data-data keuangan perusahaan.

3.2 Langkah Penelitian

Pada penelitian ini, penulis menggunakan model pengembangan oleh Luther. Model pengembangan ini terdiri dari enam tahap, yaitu *concept*, *design*, *material collecting*, *assembly*, *testing*, dan *distribution*.



Gambar 3.1 Luther Model

Tahap-tahap tersebut diimplementasikan pada penelitian yang penulis lakukan dengan rincian sebagai berikut:

1. Tahap Konsep (*Concept*)

Pada tahap ini akan menunjukkan hasil sebuah tujuan dari dibuatnya aplikasi ini. Dan berikut adalah hal yang penulis lakukan pada tahap konsep, yaitu:

a. Menentukan Tujuan

Di sini penulis menentukan tujuan dari pembuatan aplikasi ini, dan mengidentifikasi pengguna dari aplikasi ini.

Menentukan Jenis

Di sini penulis memaparkan konsep aplikasi yang akan dibuat dan mendeskripsikan jenis dan spesifikasi umum aplikasinya.

2. Tahap Perancangan (*Design*)

Saat membuat aplikasi, perlu melakukan perancangan (*design*) agar aplikasi tersebut sesuai dengan tujuan yang diinginkan. Adapun *design* yang akan dibuat, dilakukan berdasarkan langkah berikut :

a. Perancangan *Use Case Diagram*

Prancangan kebutuhan *use case diagram* ini akan menjelaskan tentang interaksi antara satu aktor atau lebih dengan sistem yang akan dibuat, dengan ini akan mengetahui tentang fungsi apa saja yang ada pada sebuah sistem dan siapa saja yang berhak untuk menggunakannya.

b. Perancangan *Activity Diagram*

Prancangan kebutuhan *activity diagram* ini akan memperlihatkan gambar proses bisnis atau aliran kerja yang terjadi pada suatu sistem. Bedanya dengan *use case diagram* adalah *activity diagram* ini menggambarkan aktifitas sistem, sedangkan *use case diagram* menjelaskan apa yang dilakukan oleh seorang aktor.

c. Perancangan *Class Diagram*

Class diagram adalah model statis yang menggambarkan deskripsi dan struktur kelas serta hubungan antar kelas.

d. Perancangan Antarmuka (*User Interface*)

User interface adalah bagaimana cara seorang pengguna akan berkomunikasi dengan sebuah program. Hal ini kadang juga sering disebut dengan istilah HCI (*Human Computer Interaction*). HCI (*Human Computer Interface*) adalah semua aspek dari interaksi pengguna dan komputer.

3. Tahap Pengumpulan Bahan (*Material Collecting*)

Tahapan pengumpulan bahan (*material collecting*) adalah tahapan dimana penulis melakukan pengumpulan bahan yang nantinya akan digunakan untuk membuat aplikasi seperti data, gambar, icon dan yang lain-lainnya.

4. Tahap Pembuatan (*Assembly*)

Setelah bahan-bahan yang dikumpulkan pada tahap *material collecting* terkumpul, maka penulis dapat menggunakan bahan tersebut untuk membuat sebuah aplikasi yang sesuai kebutuhan.

5. Tahap Pengujian (*Testing*)

Tahap pengujian (*testing*) dilakukan setelah menyelesaikan tahap pembuatan (*assembly*) dengan menjalankan aplikasi. Dalam tahap testing ini penulis menggunakan pengujian fungsional. Pengujian fungsional adalah tahap dimana aplikasi ini diuji fitur dan tombol yang ada didalamnya. Hal ini dilakukan agar penulis tahu apakah aplikasi ini sudah berjalan dengan baik atau tidak.

6. Distribusi (*Distribution*)

Pada tahap ini aplikasi yang sudah jadi akan disimpan dalam suatu media penyimpanan. Dan di tahap ini penulis juga melakukan evaluasi untuk pengembangan produk agar menjadi lebih baik. Hasil evaluasi ini dapat digunakan sebagai masukan untuk tahap *concept* pada penelitian selanjutnya.

3.3 Perancangan (*Design*)

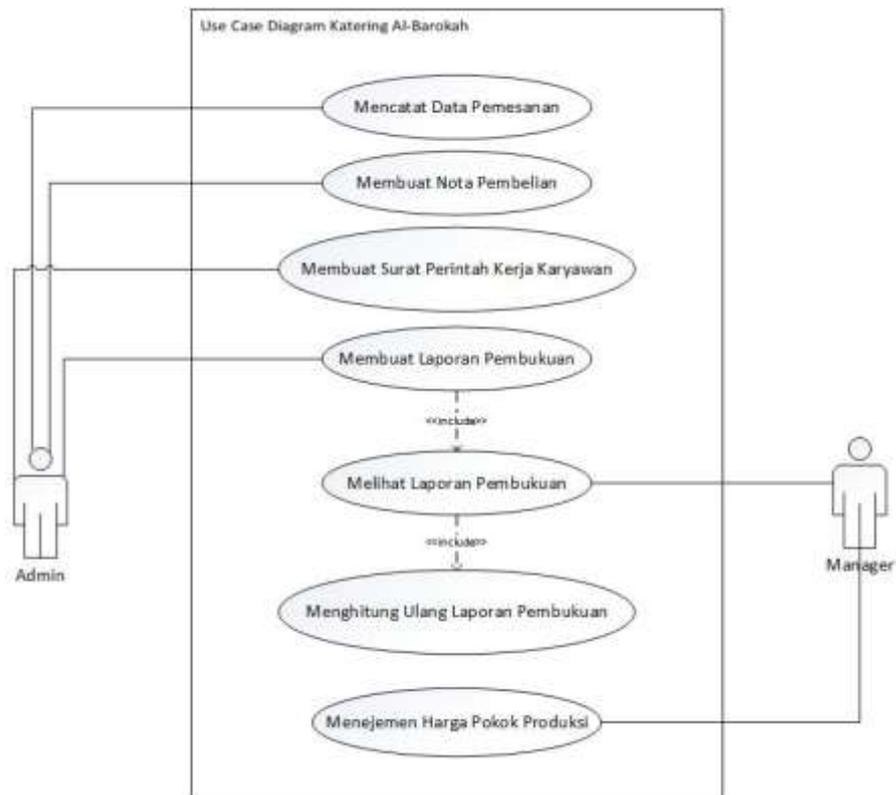
Saat membuat aplikasi, perlu melakukan perancangan (*design*) agar aplikasi tersebut sesuai dengan tujuan yang diinginkan. Rancangan yang akan dibuat, dilakukan mengikuti langkah berikut :

3.3.1 Perancangan *Use Case Diagram*

Perancangan kebutuhan *use case diagram* ini akan menjelaskan tentang interaksi antara satu aktor atau lebih dengan sistem yang akan dibuat, dengan ini akan mengetahui tentang fungsi apa saja yang ada pada sebuah sistem dan siapa saja yang berhak untuk menggunakannya.

1. Use Case Diagram Sistem Pembukuan Katering Al-Barokah

Use Case Diagram sistem pembukuan pada Katering Al-Barokah mempunyai 2 aktor dan 8 *use case*. Adapun 2 aktor ini terdiri dari admin dan manager seperti yang bisa dilihat pada gambar 3.2.



Gambar 3.2 *Use Case Diagram* Sistem Pembukuan Manual

Pada gambar 3.2 di atas admin dapat melakukan beberapa *action* seperti:

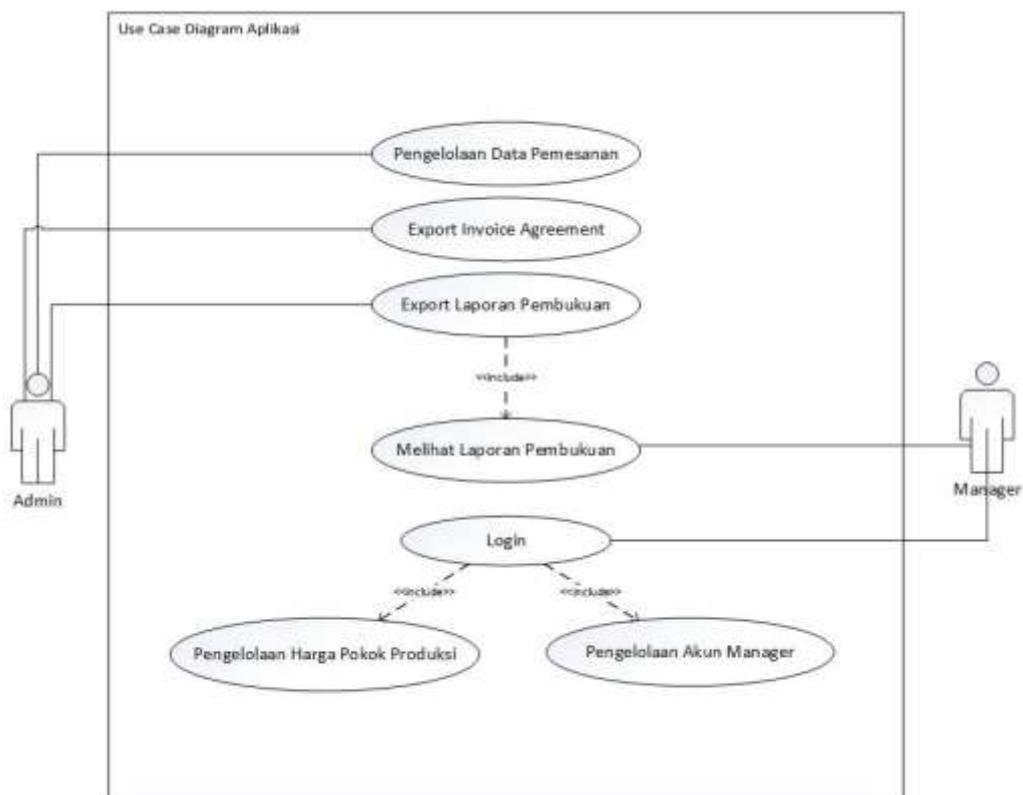
- a. Mencatat data pemesanan
- b. Membuat nota pembelian
- c. Membuat surat perintah kerja karyawan
- d. Membuat laporan pembukuan

Dan manager dapat melakukan beberapa *action* sebagai berikut:

- a. Melihat laporan pembukuan
- b. Menghitung ulang laporan pembukuan
- c. Pengelolaan harga pokok produksi

2. Use Case Diagram Aplikasi Pembukuan Katering

Use Case Diagram yang digunakan pada aplikasi “Aplikasi Pembukuan Katering” mempunyai 2 aktor dan 7 *use case*. Adapun 2 aktor ini terdiri dari admin dan manager seperti yang bisa dilihat pada gambar 3.3.



Gambar 3.3 Use Case Diagram Sistem Pembukuan Aplikasi

Pada gambar 3.3 dapat dilihat bahwa admin dapat melakukan beberapa *action* berikut:

- a. Mengelola data pemesanan
- b. *Export Invoice Agreement*
- c. *Export* laporan pembukuan

Dan manager dapat melakukan beberapa *action* seperti:

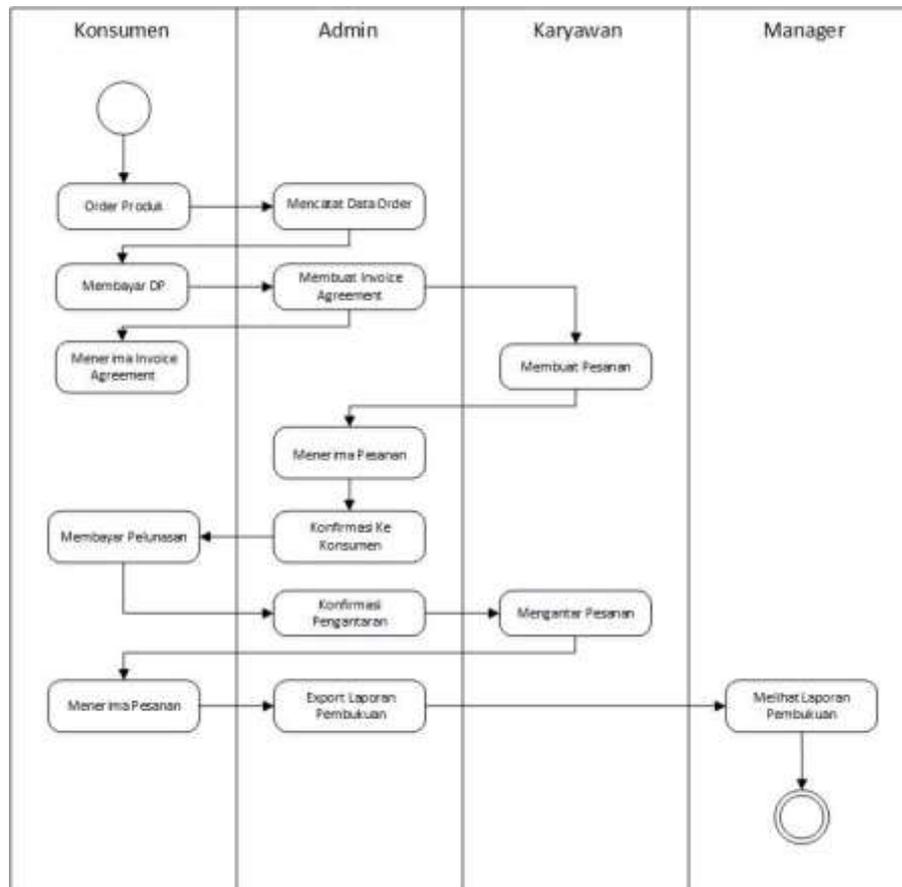
- a. Melihat laporan pembukuan
- b. *Login*
- c. Mengelola harga produk
- d. Mengelola akun manager

3.3.2 Perancangan *Activity Diagram*

Perancangan kebutuhan *activity diagram* ini akan memperlihatkan gambar proses bisnis atau aliran kerja yang terjadi pada suatu sistem. Bedanya dengan *use case diagram* adalah *activity diagram* ini menggambarkan aktifitas sistem, sedangkan *use case diagram* menjelaskan apa yang dilakukan oleh seorang aktor.

1. *Activity Diagram* Alur Pembukuan Dengan Aplikasi

Activity Diagram adalah diagram yang menggambarkan aliran kerja atau aktifitas dari sebuah proses bisnis. Dan berikut ini adalah *activity diagram* alur pembukuan Katering Al-Barokah menggunakan aplikasi yang dapat dilihat pada gambar 3.4



Gambar 3.4 Activity Diagram Alur Pembukuan Dengan Aplikasi

Keterangan Gambar 3.4 :

- a. Konsumen melakukan order produk
- b. Admin mencatat data *order*
- c. Konsumen membayar DP
- d. Admin membuat *invoice agreement*
- e. Konsumen menerima *invoice agreement*
- f. Karyawan membuat pesanan
- g. Admin menerima pesanan jadi dari karyawan
- h. Admin melakukan konfirmasi pesanan jadi ke konsumen
- i. Konsumen membayar pelunasan
- j. Admin melakukan konfirmasi pengantaran
- k. Karyawan mengantar pesanan

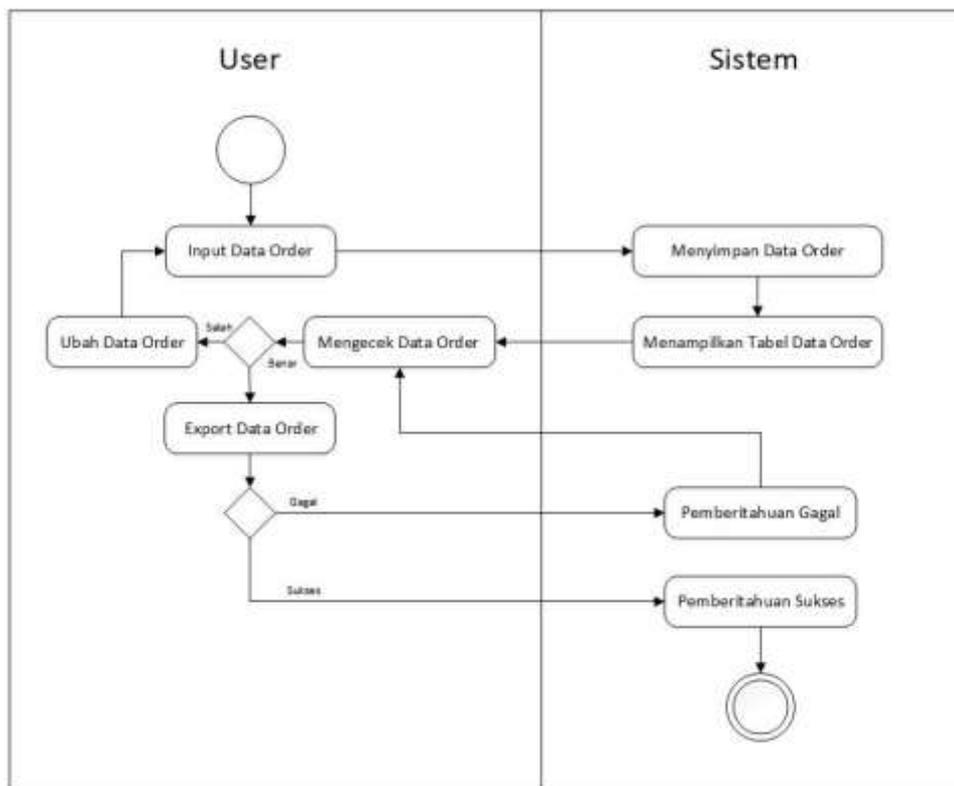
- l. Konsumen menerima pesanan
- m. Admin melakukan *export* laporan pembukuan
- n. Manager melihat laporan pembukuan

2. *Activity Diagram* Aplikasi Pembukuan

Activity Diagram merupakan alur aktifitas dalam aplikasi yang sedang dirancang. Berikut adalah *activity diagram* yang digunakan pada aplikasi pembukuan.

a. *Activity Diagram* Menu Admin

Berikut ini adalah *activity diagram* yang terdapat pada menu admin yang bisa dilihat pada gambar 3.5



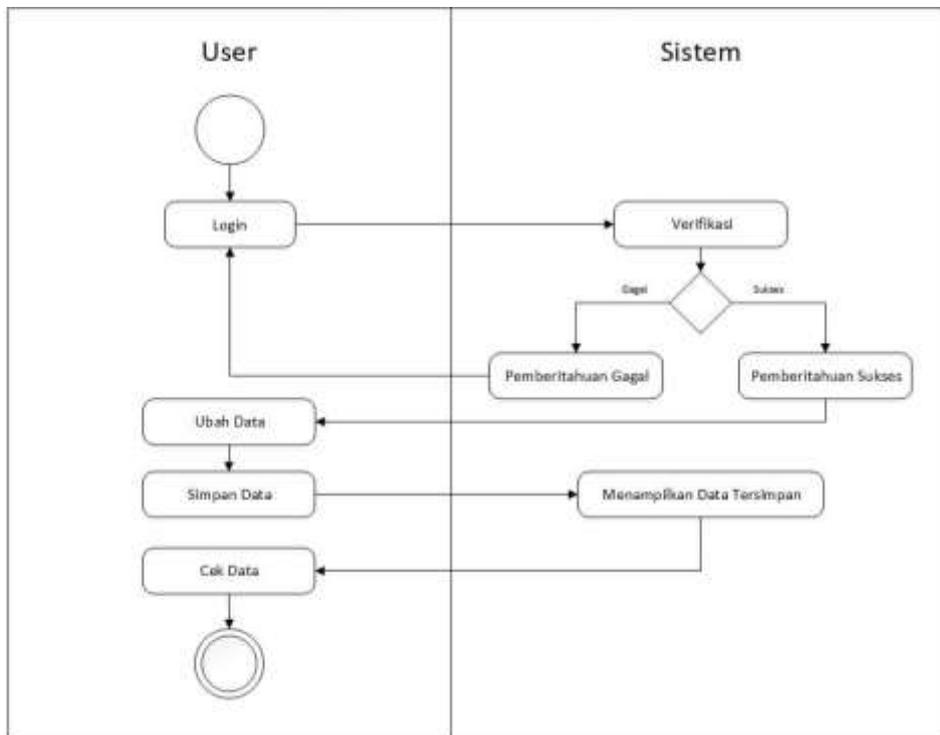
Gambar 3.5 *Activity Diagram* Menu Admin Aplikasi Pembukuan

Keterangan Gambar 3.5 :

- a. *User* melakukan *input* data
- b. Sistem menyimpan data
- c. Sistem menampilkan tabel data order yang telah disimpan
- d. *User* mengecek data order
- e. Jika salah *user* akan melakukan edit data order dan input data order kembali
- f. Jika benar *user* akan melakukan *export* data order
- g. Jika gagal sistem menampilkan pemberitahuan gagal dan *user* kembali mengecek data order
- h. Jika sukses sistem akan menampilkan pemberitahuan sukses

b. *Activity Diagram* Menu Manager

Berikut ini *activity diagram* menu manager dapat dilihat pada gambar 3.6

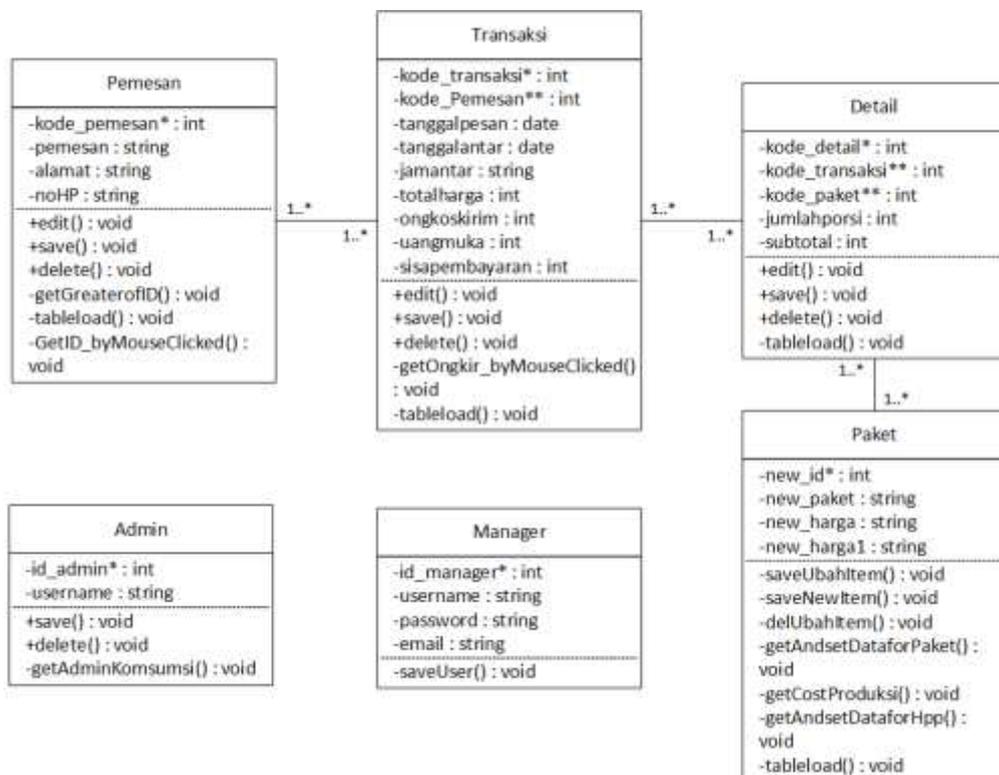


Gambar 3.6 *Activity Diagram* Menu Manager Aplikasi Pembukuan

- a. *User* melakukan *login*
- b. Sistem memverifikasi
- c. Jika gagal sistem akan menampilkan pemberitahuan gagal dan *user* kembali *login*
- d. Jika sukses sistem akan menampilkan pemberitahuan sukses
- e. *User* mengubah data
- f. *User* menyimpan data
- g. Sistem menampilkan data yang telah tersimpan
- h. *User* mengecek data tersimpan

3.3.3 Perancangan *Class Diagram*

Class Diagram adalah model statis yang menggambarkan deskripsi dan struktur kelas serta hubungan antar kelas. Dan berikut adalah *class diagram* pada aplikasi pembukuan catering yang dapat dilihat pada gambar 3.7



Gambar 3.7 *Class Diagram* Aplikasi Pembukuan

Berikut penjelasan terkait gambar 3.7 mengenai *class diagram* diatas:

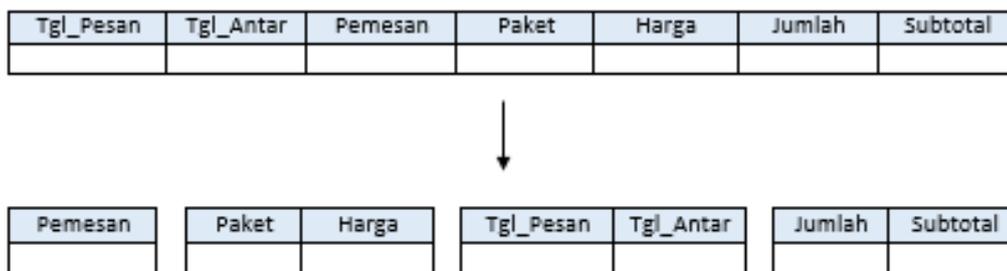
1. *Class Admin* mempunyai fungsi menyimpan seluruh data admin yang ada pada sistem, seperti *Id_Admin*, *Username*.
2. *Class Manager* mempunyai fungsi menyimpan seluruh data manager yang ada pada sistem, seperti *Id_Manager*, *Username*, *Password*, dan *Email*.
3. *Class Paket* mempunyai fungsi menyimpan seluruh data harga produksi yang ada pada sistem, seperti *Kode_Paket*, *Paket*, *Harga*.
4. *Class Pemesan* mempunyai fungsi menyimpan seluruh data pemesan yang ada pada sistem, seperti *Id_Pemesan*, *Nama*, *Alamat*, dan *No_Handphone*.
5. *Class Transaksi* mempunyai fungsi menyimpan seluruh data transaksi pemesanan yang ada pada sistem, seperti *Kode_Transaksi*, *Id_Pemesan*, *Tanggal_Pesan*, *Tanggal_Antar*, *Jam_Antar*, *Total*, *Ongkir*, *Uang_Muka*, *Sisa_Bayar*.
6. *Class Detail* mempunyai fungsi menyimpan seluruh data detail transaksi yang ada pada sistem, seperti *Kode_Detail*, *Kode_Transaksi*, *Kode_Paket*, *Harga_Jual*, *Jumlah*, dan *Subtotal*.

3.3.4 Normalisasi Tabel

Normalisasi *database* digunakan untuk menghilangkan dan mengurangi redundansi data dan tujuan lainnya yaitu untuk memastikan dependensi data (data berada pada tabel yang tepat) agar database tersebut dapat bekerja dengan optimal. Berikut merupakan gambar tabel yang perlu di normalisasi.

1. Normalisasi Pertama

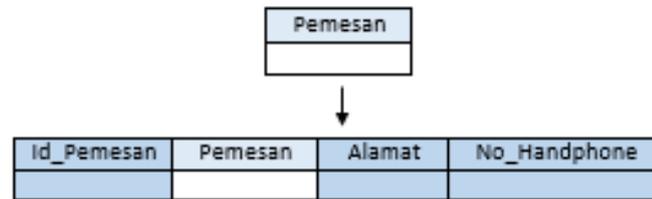
ANGKAL PESAN	TANGGAL ANTAR	PEMESAN	PAKET	JUMLAH	HARGA PER BOX	TOTAL
03-08-18	06-08-18	MAS ARUL	B	35	12.500	437.500
07-08-18	11-08-18	Muti Sudi	A	20	10.000	200.000
09-08-18	11-08-18	SD H Kaffan	A	50	10.000	500.000
11-08-18	14-08-18	Muti Irma	C	32	14.000	448.000
13-08-18	15-08-18	Cater Dita Tamonika	C	112	14.000	1.568.000
16-08-18	22-08-18	Muti Sutadjo	A	25	10.000	250.000
20-08-18	22-08-18	HMS UMY	A	30	10.000	300.000
25-08-18	27-08-18	SPBU Ambarharjo	A	18	10.000	180.000
25-08-18	28-08-18	Bu Siti	B	41	12.500	512.500
28-08-18	30-08-18	Muti Al-Falah	C	42	14.000	588.000
29-08-18	30-08-18	KMTI UMY	B	35	12.500	437.500
30-08-18	01-09-18	TK bina Iqroni	A	40	10.000	400.000
30-08-18	02-09-18	Paik Farid	C	100	14.000	1.400.000



Gambar 3.8 Normalisasi Pertama

Penjelasan pada gambar 3.8 diatas adalah tabel catering dinormalisasi yang berfungsi membuat tabel terpisah untuk masing-masing kelompok data yang berkaitan. Dari normal form lalu dipisah menjadi 3 tabel agar mengurangi kesalahan-kesalahan yang dimungkinkan.

a. Tabel Pemesan



Gambar 3.9 Tabel Pemesan

Pada tabel pemesan terjadi penambahan kolom baru yang bertujuan sebagai penambah informasi agar data yang dihasilkan menjadi lebih lengkap. Berikut beberapa kolom yang ditambahkan adalah kolom *Id_Pemesan* sebagai *primary key*, kolom *Alamat* sebagai informasi alamat pemesan sekaligus alamat tujuan pengiriman, dan kolom *No_Handphone* sebagai informasi nomor *handphone* pemesan.

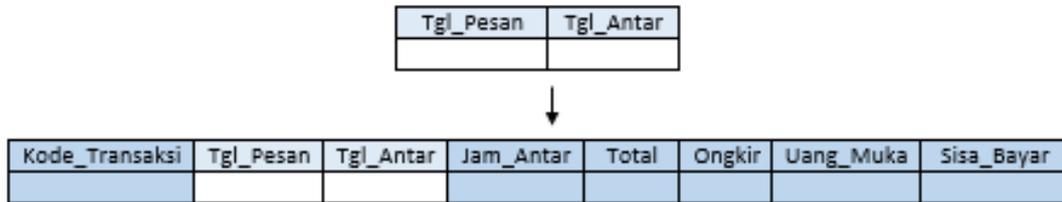
b. Tabel Paket



Gambar 3.10 Tabel Paket

Pada tabel paket terjadi penambahan kolom baru yang bertujuan sebagai penambah informasi agar data yang dihasilkan menjadi lebih lengkap. Berikut beberapa kolom yang ditambahkan adalah kolom *Kode_Paket* sebagai *primary key*, kolom *Harga_Produksi* sebagai informasi harga produksi paket.

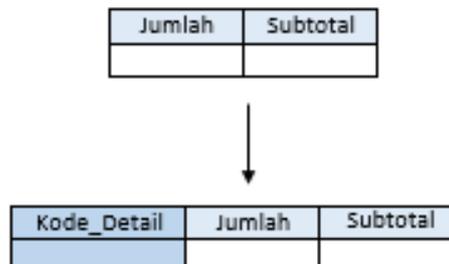
c. Tabel Transaksi



Gambar 3.11 Tabel Transaksi

Pada tabel paket terjadi penambahan kolom baru yang bertujuan sebagai penambah informasi agar data yang dihasilkan menjadi lebih lengkap. Berikut beberapa kolom yang ditambahkan adalah kolom Kode_Transaksi sebagai *primary key*, kolom Jam_Antar sebagai informasi waktu pengantaran.

d. Tabel Detail

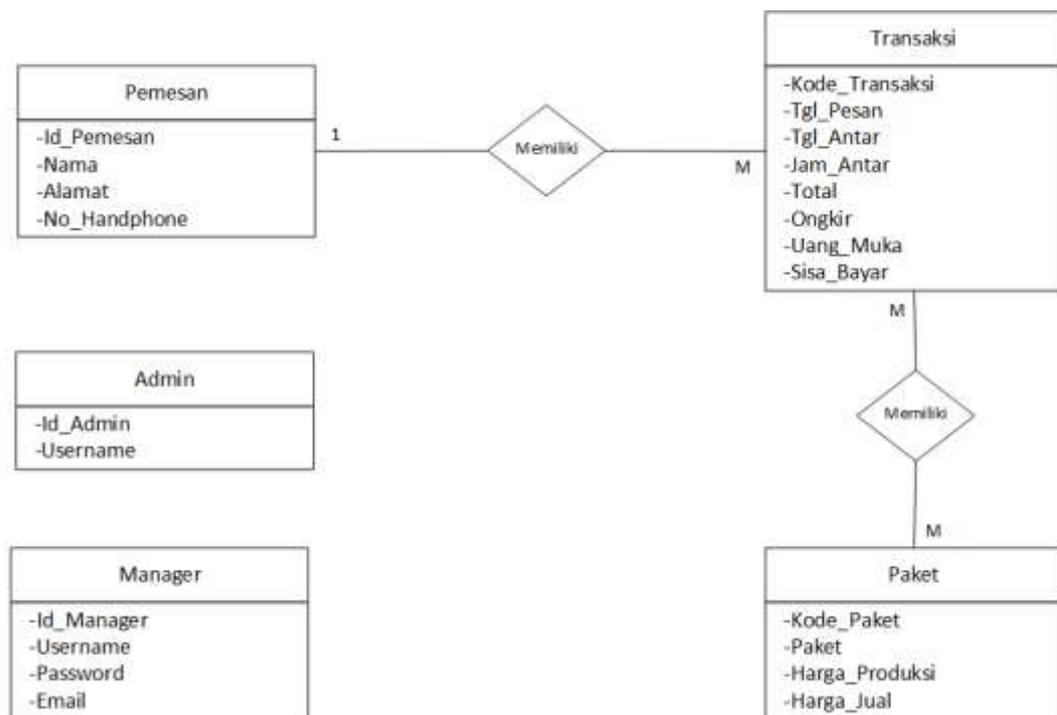


Gambar 3.12 Tabel Detail

Pada tabel detail terjadi penambahan kolom baru yang bertujuan sebagai penambah informasi agar data yang dihasilkan menjadi lebih lengkap. Berikut kolom yang ditambahkan adalah kolom Kode_Detail sebagai *primary key*.

3.3.5 Perancangan Entity Relationship Diagram

Entity Relationship Diagram adalah suatu model untuk menjelaskan hubungan antar data dalam basis data berdasarkan objek-objek dasar data yang mempunyai hubungan antar relasi. Dan berikut adalah *entity relationship diagram* pada aplikasi pembukuan catering yang dapat dilihat pada gambar 3.13.



Gambar 3.13 *Entity Relationship Diagram* Aplikasi Pembukuan

Penjelasan ERD pada gambar 3.13 adalah sebagai berikut:

1. *Entitas Relationship Diagram* aplikasi pembukuan catering memiliki 6 buah entitas yaitu:
 - a. Admin
 - b. Pemesan
 - c. Transaksi
 - d. Paket
 - e. Manager

2. Entitas admin merupakan entitas yang kuat karena entitas admin tidak bergantung pada entitas lain.
3. Entitas pemesan dengan entitas transaksi menggunakan relasi *one-to-many*.
4. Entitas transaksi dengan entitas paket menggunakan relasi *many-to-many*.
5. Entitas manager merupakan entitas yang kuat karena entitas manager tidak bergantung pada entitas lain.
6. Entitas detail merupakan tabel baru yang terbentuk dari relasi antara entitas transaksi dengan entitas paket

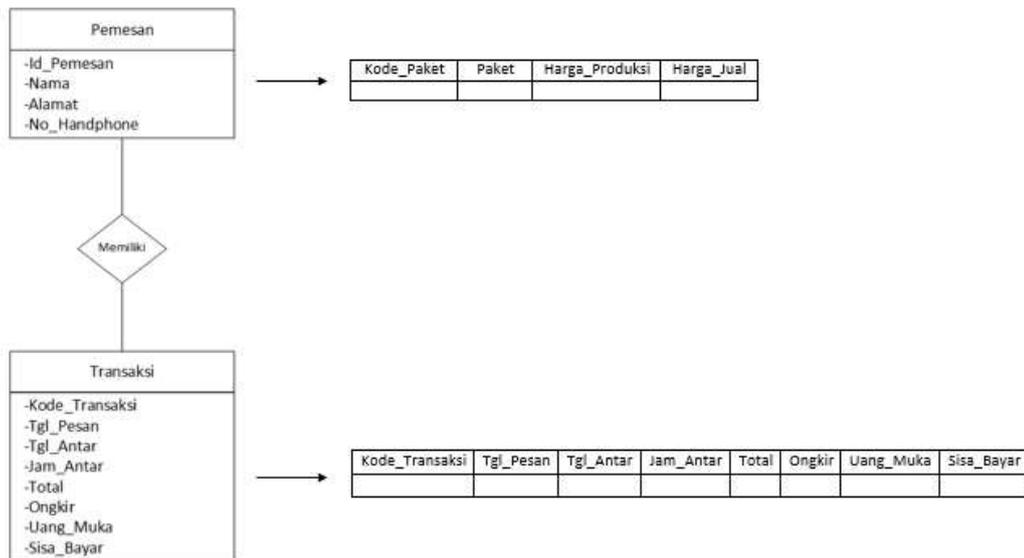
3.3.6 Konversi ERD ke dalam Tabel

Konversi ERD ke dalam bentuk tabel adalah proses memasukkan entitas menjadi nama tabel dan *attributes* nama kolom beserta relasinya ke dalam sebuah tabel yang saling berhubungan.

1. Konversi dengan relasi 1-M (*one to many*)

Konversi entitas yang memiliki relasi 1-M (*one to many*) adalah setiap baris data pada tabel pertama dihubungkan satu baris atau lebih dari satu baris data pada tabel kedua. Pada ERD yang ditunjukkan pada gambar 3.13 ditunjukkan terdapat relasi *one to many* yaitu antara entitas pemesan dengan entitas transaksi. Dari entitas tersebut akan direpresentasikan kedalam tabel, dapat dilihat pada gambar 3.14 sebagai berikut.

Entitas Pemesan dengan Entitas Transaksi



Gambar 3.14 Relasi Entitas Pemesan dengan Entitas Transaksi

Penjelasan konversi relasi entitas pemesan dengan entitas transaksi pada gambar 3.14 adalah sebagai berikut:

Entitas pemesan memiliki relasi *one to many* dengan entitas transaksi yang artinya satu pemesan dapat memiliki lebih dari satu transaksi maka kolom Id_Pemesan yang ada pada tabel pemesan perlu ditambahkan ke tabel transaksi yang berfungsi sebagai *foreign key* pada tabel transaksi.

2. Konversi dengan relasi M-M (*many to many*)

Konversi entitas yang memiliki relasi M-M (*many to many*) adalah menghubungkan dua entitas, diwujudkan dalam bentuk tabel khusus yang memiliki *field (foreign key)* yang berasal dari kunci-kunci (*key*) dari entitas yang dihubungkan sehingga menciptakan tabel baru. Pada ERD yang ditunjukkan pada gambar 3.13 ditunjukkan terdapat relasi *many to many* yaitu antara entitas transaksi dengan entitas paket. Dari entitas tersebut akan direpresentasikan kedalam tabel, dapat dilihat pada gambar 3.15 sebagai berikut.

Entitas Transaksi dengan Entitas Paket



Gambar 3.15 Relasi Entitas Transaksi dengan Entitas Paket

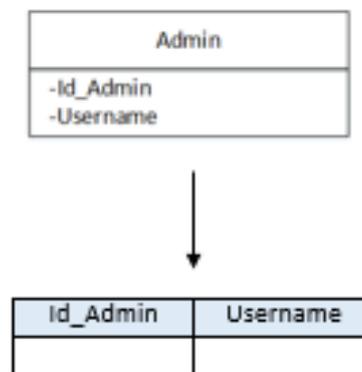
Penjelasan konversi relasi entitas transaksi dengan entitas paket pada gambar 3.15 adalah sebagai berikut:

Entitas transaksi memiliki relasi *many to many* dengan entitas paket sehingga tercipta tabel baru yang menghubungkan entitas transaksi dengan entitas paket yang diberi nama tabel detail, dimana kolom Kode_Transaksi dan Kode_Paket perlu ditambahkan pada tabel detail sebagai *foreign key*.

3. Konversi Entitas Kuat

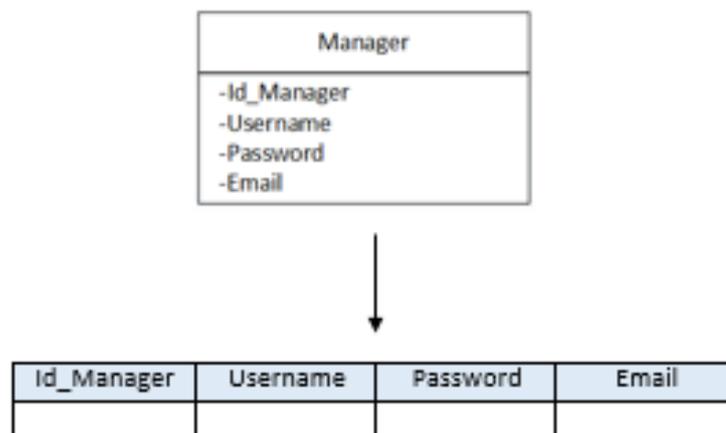
Konversi entitas kuat adalah entitas yang dilibatkan atau dicantumkan secara jelas dalam ERD, dan tidak memiliki ketergantungan terhadap entitas lain. Pada gambar 3.13 terdapat entitas kuat yaitu entitas admin dan entitas manager yang dapat dilihat pada gambar 3.16 dan gambar 3.17.

a. Entitas Kuat Admin



Gambar 3.16 Konversi Entitas Kuat Admin

b. Entitas Kuat Manager

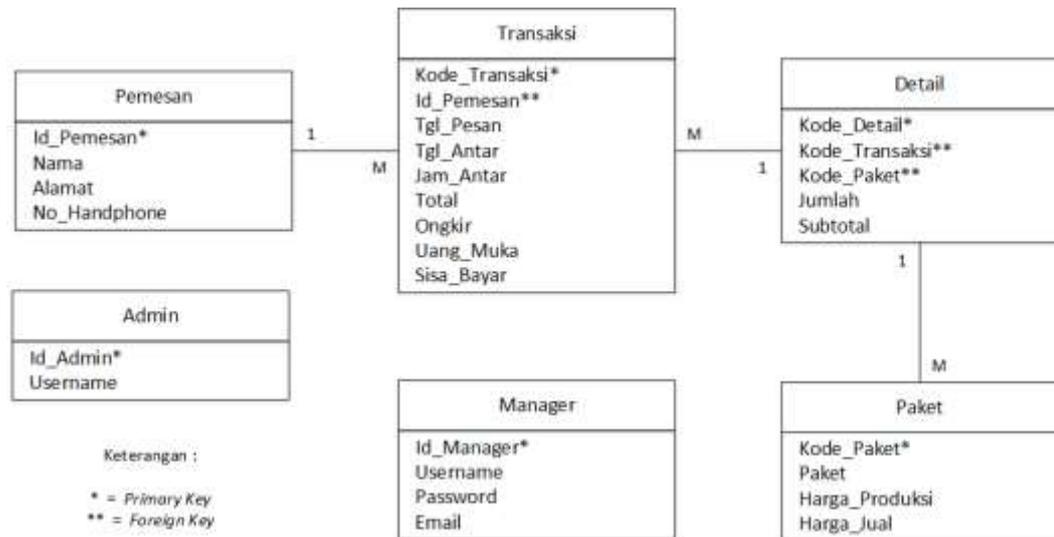


Gambar 3.17 Entitas Manager

Pada proses konversi entitas admin dan entitas manager dirubah ke dalam bentuk tabel pada gambar 3.16 dan gambar 3.17 dijelaskan bahwa entitas menjadi nama tabel dan *attributes* menjadi nama kolom.

3.3.7 Relasi Antar Tabel

Setelah dilakukan konfersi ERD ke dalam tabel maka akan didapatkan relasi antar tabel yang dapat dilihat pada gambar 3.18 sebagai berikut.



Gambar 3.18 Relasi antar Tabel

Penjelasan relasi antar tabel yang telah digambarkan seperti pada gambar 3.18 diatas adalah sebagai berikut:

- Tabel pemesanan memiliki relasi *one to many* dengan tabel transaksi untuk menghubungkan tabel pemesanan dengan tabel transaksi maka kolom Id_Pemesan harus ditambahkan sebagai *foreign key* pada tabel transaksi.
- Tabel transaksi memiliki relasi *one to many* dengan tabel detail untuk menghubungkan tabel transaksi dengan tabel detail maka kolom Kode_Transaksi harus ditambahkan sebagai *foreign key* pada tabel detail.
- Tabel paket memiliki relasi *one to many* dengan tabel detail untuk menghubungkan tabel paket dengan tabel detail maka kolom Kode_Paket harus ditambahkan sebagai *foreign key* pada tabel detail.

3.3.8 Kamus Database

Berikut ini merupakan tabel-tabel dari kamus database pada “Aplikasi Pembukuan Katering Berbasis Desktop”.

1. Kamus Data Pemesan

Kamus data tabel pemesanan yang digunakan dalam aplikasi desktop dapat dilihat pada tabel 3.1.

Tabel 3.1 Kamus Data Tabel Pemesan

Nama Kolom	Tipe Data	Panjang Data	Key	Keterangan
Id_Pemesan	Int	-	<i>Primary Key</i>	ID Pemesan
Nama	Varchar	25		Nama Pemesan
Alamat	Varchar	50		Alamat
No_Handphone	Varchar	13		Nomor HP Pemesan

2. Kamus Data Manager

Kamus data tabel admin yang digunakan dalam aplikasi desktop dapat dilihat pada tabel 3.2.

Tabel 3.2 Kamus Data Tabel Manager

Nama Kolom	Tipe Data	Panjang Data	Key	Keterangan
Id_Manager	Int	-	<i>Primary Key</i>	ID Manager
Username	Varchar	10		Nama Manager
Password	Varchar	10		Kata Sandi
Email	Varchar	30		Email Manager

3. Kamus Data Admin

Kamus data tabel admin yang digunakan dalam aplikasi desktop dapat dilihat pada tabel 3.3.

Tabel 3.3 Kamus Data Tabel Admin

Nama Kolom	Tipe Data	Panjang Data	Key	Keterangan
Id_Admin	Int	-	<i>Primary Key</i>	ID Admin
Username	Varchar	10		Nama Admin

4. Kamus Data Transaksi

Kamus data tabel transaksi yang digunakan dalam aplikasi desktop dapat dilihat pada tabel 3.4.

Tabel 3.4 Kamus Data Tabel Transaksi

Nama Kolom	Tipe Data	Panjang Data	Key	Keterangan
Kode_Transaksi	Int	-	<i>Primary Key</i>	Kode Transaksi
Id_Pemesan	Int	-	<i>Foreign Key</i>	ID Pemesan
Tgl_Pesan	Date	-		Tanggal Pesan
Tgl_Antar	Date	-		Tanggal Antar
Jam_Antar	Varchar	5		Jam Antar
Total	Real	-		Jumlah Seluruh Paket Dipesan
Ongkir	Real	-		Ongkos Kirim
Uang_Muka	Real	-		Uang Muka
Sisa_Bayar	Real	-		Sisa Bayar (Total + Ongkir - DP)

5. Kamus Data Paket

Kamus data tabel manager yang digunakan dalam aplikasi desktop dapat dilihat pada tabel 3.5.

Tabel 3.5 Kamus Data Tabel Paket

Nama Kolom	Tipe Data	Panjang Data	Key	Keterangan
Kode_Paket	Int	-	<i>Primary Key</i>	Kode Paket
Paket	Varchar	5		Jenis Paket
Harga_Produksi	Real	-		Harga Produksi
Harga_Jual	Real	-		Harga Jual

6. Kamus Data Detail

Kamus data tabel detail yang digunakan dalam aplikasi desktop dapat dilihat pada tabel 3.6.

Tabel 3.6 Kamus Data Tabel Detail

Nama Kolom	Tipe Data	Panjang Data	Key	Keterangan
Kode_Detail	Int	-	<i>Primary Key</i>	Kode Detail
Kode_Transaksi	Int	-	<i>Foreign Key</i>	Kode Transaksi
Kode_Paket	Int	-	<i>Foreign Key</i>	Kode Paket
Jumlah	Int	-		Jumlah Pesanan
Subtotal	Real	-		Total Harga (Harga Paket x Jumlah)

3.3.9 Perancangan *User Interface*

Perancangan user interface merupakan tampilan dari suatu aplikasi yang bertujuan sebagai media komunikasi antara perangkat lunak dan manusia. Rancangan ini merupakan gambaran sederhana dari tampilan yang akan diimplementasikan dalam penelitian. Berikut adalah rancangan antarmuka pada aplikasi pembukuan catering.

1. Perancangan *User Interface* Halaman Utama

Berikut ini merupakan perancangan user interface halaman utama aplikasi pembukuan catering. Halaman menu utama terdapat 2 buah menu pilihan yaitu admin dan manager, seperti yang terlihat pada gambar 3.19



Gambar 3.19 Perancangan *User Interface* Menu Utama

2. Perancangan *User Interface* Menu Admin

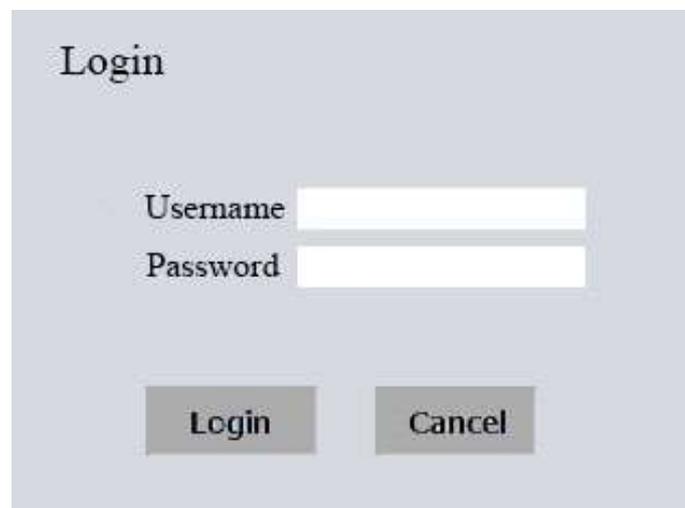
Pada perancangan *User Interface* menu admin akan ditampilkan beberapa *button*, *textfield*, *combobox* serta *table* yang akan digunakan untuk melakukan *input*, *edit* serta menampilkan data, seperti yang terlihat pada gambar 3.20



Gambar 3.20 Perancangan *User Interface* Menu Admin

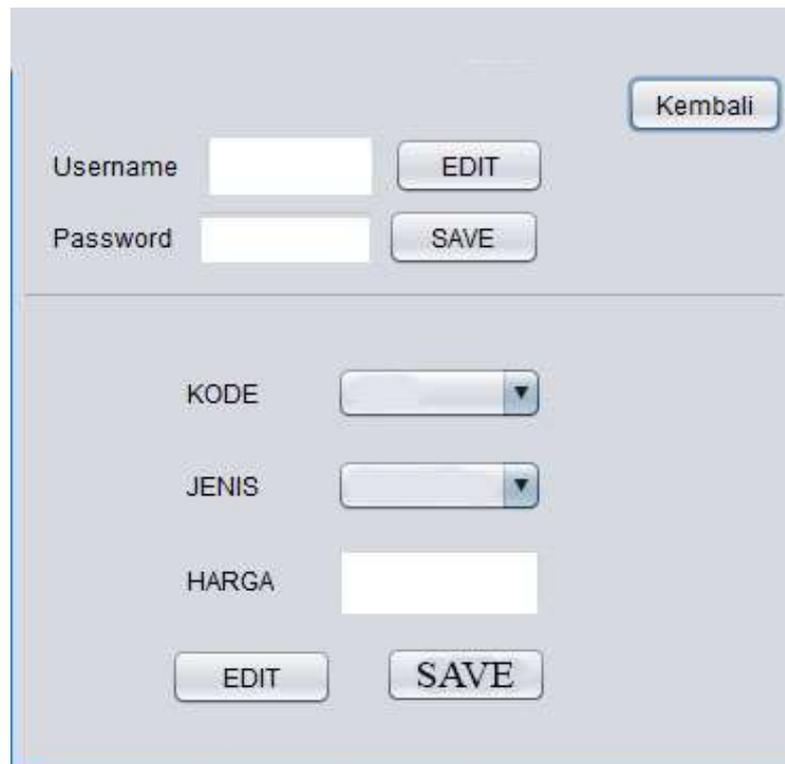
3. Perancangan *User Interface* Menu Manager

Pada perancangan *User Interface* menu manager akan menampilkan halaman *login* yang berisi beberapa *button*, *textfield*, dan *label* sebelum *user* masuk pada tampilan inti seperti yang terlihat pada gambar 3.21



Gambar 3.21 Perancangan User Interface Halaman *Login*

Pada tampilan inti dalam menu manager setelah *login* akan menampilkan beberapa *button*, *textfield*, dan *label* yang akan digunakan untuk melakukan *input*, *edit* data, seperti terlihat pada gambar 3.22.



Gambar 3.22 Perancangan *User Interface* Menu HPP

4. Perancangan *User Interface* Laporan Pembukuan Excel

Pada perancangan *User Interface* laporan pembukuan excel akan menampilkan 5 *sheet*. Adapun itu terdiri dari *sheet* data pemesanan, dana gaji karyawan, kas perusahaan, *sponsorship*, dan sedekah sebagai berikut:

a. Data Pemesanan

Pada *sheet* ini akan berisi *cashflow* dari proses bisnis yang terjadi dimana ada debit, kredit, omset, serta pemasukan bersih dari proses bisnis yang berlangsung seperti pada gambar 3.23.

Catatan Pemesanan dan Keuangan Bulan Agustus 2018							Pembukuan			
No.	Tanggal	Pemesan	Paket	Jumlah	Harga Produk	Harga Jual	Total H Produk	Total H Jual		
1							Rp	-	Rp	-
2							Rp	-	Rp	-
3							Rp	-	Rp	-
4							Rp	-	Rp	-
5							Rp	-	Rp	-
6							Rp	-	Rp	-
7							Rp	-	Rp	-
8							Rp	-	Rp	-
9							Rp	-	Rp	-
10							Rp	-	Rp	-
11							Rp	-	Rp	-
12							Rp	-	Rp	-
13							Rp	-	Rp	-
14							Rp	-	Rp	-
15							Rp	-	Rp	-
16							Rp	-	Rp	-
17							Rp	-	Rp	-
18							Rp	-	Rp	-
19							Rp	-	Rp	-
20							Rp	-	Rp	-
21							Rp	-	Rp	-
22							Rp	-	Rp	-
23							Rp	-	Rp	-
24							Rp	-	Rp	-
25							Rp	-	Rp	-

Omzet	Rp	-
- Biaya Produk	Rp	-
Total Pemesukan	Rp	-
Total Semesta	Rp	-
Pemesukan Bersih	Rp	-

Gambar 3.23 User Interface Sheet Data Pemesanan

b. Dana Gaji Karyawan

Pada sheet ini akan berisi jumlah persenan dari omzet perusahaan yang dialokasikan untuk menggaji karyawan yang bisa dilihat seperti pada gambar 3.24.

Catatan Gaji Karyawan Bulan Agustus 2018						Total Dana	Rp	-
No	Tanggal	Pemesan	Paket	Jumlah	Total Cost			
25	00-Jan-00	0	0	0	Rp	-		
26	00-Jan-00	0	0	0	Rp	-		
27	00-Jan-00	0	0	0	Rp	-		
28	00-Jan-00	0	0	0	Rp	-		
29	00-Jan-00	0	0	0	Rp	-		
30	00-Jan-00	0	0	0	Rp	-		
31	00-Jan-00	0	0	0	Rp	-		
32	00-Jan-00	0	0	0	Rp	-		
33	00-Jan-00	0	0	0	Rp	-		
34	00-Jan-00	0	0	0	Rp	-		
35	00-Jan-00	0	0	0	Rp	-		
36	00-Jan-00	0	0	0	Rp	-		
37	00-Jan-00	0	0	0	Rp	-		
38	00-Jan-00	0	0	0	Rp	-		
39	00-Jan-00	0	0	0	Rp	-		
40	00-Jan-00	0	0	0	Rp	-		
41	00-Jan-00	0	0	0	Rp	-		
42	00-Jan-00	0	0	0	Rp	-		
43	00-Jan-00	0	0	0	Rp	-		
44	00-Jan-00	0	0	0	Rp	-		
45	00-Jan-00	0	0	0	Rp	-		

Gambar 3.24 User Interface Sheet Gaji Karyawan

c. Kas Perusahaan

Pada *sheet* ini akan berisi jumlah pesanan dari omset perusahaan yang dialokasikan untuk kas perusahaan atau simpanan perusahaan yang bisa dilihat seperti pada gambar 3.25.

Catanan Kas Perusahaan Agustus 2018						Total Dana	Rp	-
No	Tanggal	Pemesan	Paket	Jumlah	Total Cost			
1	00-Jan-00	0	0	0	Rp -			
2	00-Jan-00	0	0	0	Rp -			
3	00-Jan-00	0	0	0	Rp -			
4	00-Jan-00	0	0	0	Rp -			
5	00-Jan-00	0	0	0	Rp -			
6	00-Jan-00	0	0	0	Rp -			
7	00-Jan-00	0	0	0	Rp -			
8	00-Jan-00	0	0	0	Rp -			
9	00-Jan-00	0	0	0	Rp -			
10	00-Jan-00	0	0	0	Rp -			
11	00-Jan-00	0	0	0	Rp -			
12	00-Jan-00	0	0	0	Rp -			
13	00-Jan-00	0	0	0	Rp -			
14	00-Jan-00	0	0	0	Rp -			
15	00-Jan-00	0	0	0	Rp -			

Gambar 3.25 User Interface Sheet Kas Perusahaan

d. Sponsorship

Pada *sheet* ini akan berisi jumlah pesanan dari omset perusahaan yang dialokasikan untuk dana *sponsorship* yang bisa dilihat seperti pada gambar 3.26.

Catanan Sponsorship Bulan Agustus 2018						Total Dana	Rp	-
No	Tanggal	Pemesan	Paket	Jumlah	Total Cost			
1	00-Jan-00	0	0	0	Rp -			
2	00-Jan-00	0	0	0	Rp -			
3	00-Jan-00	0	0	0	Rp -			
4	00-Jan-00	0	0	0	Rp -			
5	00-Jan-00	0	0	0	Rp -			
6	00-Jan-00	0	0	0	Rp -			
7	00-Jan-00	0	0	0	Rp -			
8	00-Jan-00	0	0	0	Rp -			
9	00-Jan-00	0	0	0	Rp -			
10	00-Jan-00	0	0	0	Rp -			
11	00-Jan-00	0	0	0	Rp -			
12	00-Jan-00	0	0	0	Rp -			
13	00-Jan-00	0	0	0	Rp -			
14	00-Jan-00	0	0	0	Rp -			
15	00-Jan-00	0	0	0	Rp -			

Gambar 3.26 User Interface Sheet Sponsorship

e. Sedekah

Pada *sheet* ini akan berisi jumlah pesanan dari omset perusahaan yang dialokasikan untuk dana sedekah yang bisa dilihat seperti pada gambar 3.27.

Catatan Sedekah Bulan Agustus 2018									
No	Tanggal	Pemesan	Paket	Jumlah	Total Cost	Total Dana	Rp	-	
1	00-Jan-00	0	0	0	Rp -				
2	00-Jan-00	0	0	0	Rp -				
3	00-Jan-00	0	0	0	Rp -				
4	00-Jan-00	0	0	0	Rp -				
5	00-Jan-00	0	0	0	Rp -				
6	00-Jan-00	0	0	0	Rp -				
7	00-Jan-00	0	0	0	Rp -				
8	00-Jan-00	0	0	0	Rp -				
9	00-Jan-00	0	0	0	Rp -				
10	00-Jan-00	0	0	0	Rp -				
11	00-Jan-00	0	0	0	Rp -				
12	00-Jan-00	0	0	0	Rp -				
13	00-Jan-00	0	0	0	Rp -				
14	00-Jan-00	0	0	0	Rp -				
15	00-Jan-00	0	0	0	Rp -				

Gambar 3.27 *User Interface Sheet Sedekah*

5. Perancangan *User Interface* Kwitansi Excel

Pada perancangan *User Interface* kwitansi akan menampilkan rincian dari data pemesanan yang nantinya akan disepakati oleh pemesan seperti jenis pesanan, jam antar pesanan, jumlah, dll seperti yang terlihat pada gambar 3.28.

Al-Barokah Katering						
KWITANSI						
NO. INVOICE		NAMA				
TANGGAL PESAN		NO. HANDPHONE				
TANGGAL ANTAR		JAM ANTAR				
PAKET		JUMLAH				
ALAMAT						
RINCIAN BIAYA						
NO	JENIS PAKET	JUMLAH	ONGKIR	BIAYA		
1						
TOTAL BIAYA						
UANG MUKA						
SISA PEMBAYARAN						
<p style="margin: 0;">YOGYAKARTA,</p> <table style="width: 100%; border: none;"> <tr> <td style="width: 50%; text-align: center; vertical-align: top;"> <p style="margin: 0;">CUSTOMER SERVICE</p> <p style="margin: 0;">()</p> </td> <td style="width: 50%; text-align: center; vertical-align: top;"> <p style="margin: 0;">PELANGGAN</p> <p style="margin: 0;">()</p> </td> </tr> </table>					<p style="margin: 0;">CUSTOMER SERVICE</p> <p style="margin: 0;">()</p>	<p style="margin: 0;">PELANGGAN</p> <p style="margin: 0;">()</p>
<p style="margin: 0;">CUSTOMER SERVICE</p> <p style="margin: 0;">()</p>	<p style="margin: 0;">PELANGGAN</p> <p style="margin: 0;">()</p>					

Gambar 3.28 User Interface Kwitansi

3.4 Tahap Pengumpulan Bahan (*Material Collecting*)

Material Collecting atau dalam Bahasa Indonesia disebut dengan tahap pengumpulan bahan adalah kegiatan yang dilakukan penulis untuk pengumpulan bahan yang akan digunakan untuk pembuatan aplikasi seperti gambar, *icon*, dan data perusahaan. Bahan-bahan yang digunakan dalam membuat aplikasi pembukuan katering ini didapat dari berbagai sumber, dan sebagian besar diedit kembali oleh penulis dengan menggunakan aplikasi CorelDRAW X7.

3.4.1 File Gambar

File gambar yang ada pada aplikasi ini, penulis dapat dari perusahaan terkait dan diedit dengan menggunakan *software* CorelDRAW X7 dan di simpan dalam bentuk file PNG.

Deskripsi komponen atau item gambar yang akan digunakan untuk pembuatan aplikasi pembukuan katering dapat dilihat pada tabel 3.7.

Tabel 3.7 Deskripsi Komponen Gambar Aplikasi

Bentuk Gambar	Deskripsi
	Logo Unit Bisnis
<i>Al-Barokah Katering</i>	Nama Unit Bisnis

3.4.2 File Icon

File *icon* yang ada dalam aplikasi ini didapat dari aplikasi Icon8 dan diedit kembali oleh penulis dengan menggunakan *software* CorelDRAW X7 dan disimpan dalam bentuk PNG.

Deskripsi *icon* yang akan digunakan untuk pembuatan aplikasi pembukuan katering dapat dilihat pada tabel 3.8.

Tabel 3.8 Deskripsi Komponen Icon Aplikasi

Bentuk Icon	Deskripsi
	<i>Icon Button Add</i>
	<i>Icon Button Save</i>
	<i>Icon Button Delete</i>
	<i>Icon Button Edit Akun Manager</i>
	<i>Icon Button Kembali ke Home</i>
	<i>Icon Button Tambah Produk</i>
	<i>Icon Button Clear</i>
	<i>Icon Button Input</i>
	<i>Icon Button Browse</i>
	<i>Icon Button Export Book</i>
	<i>Icon Button Export Agreement</i>

3.4.3 Data *Input*

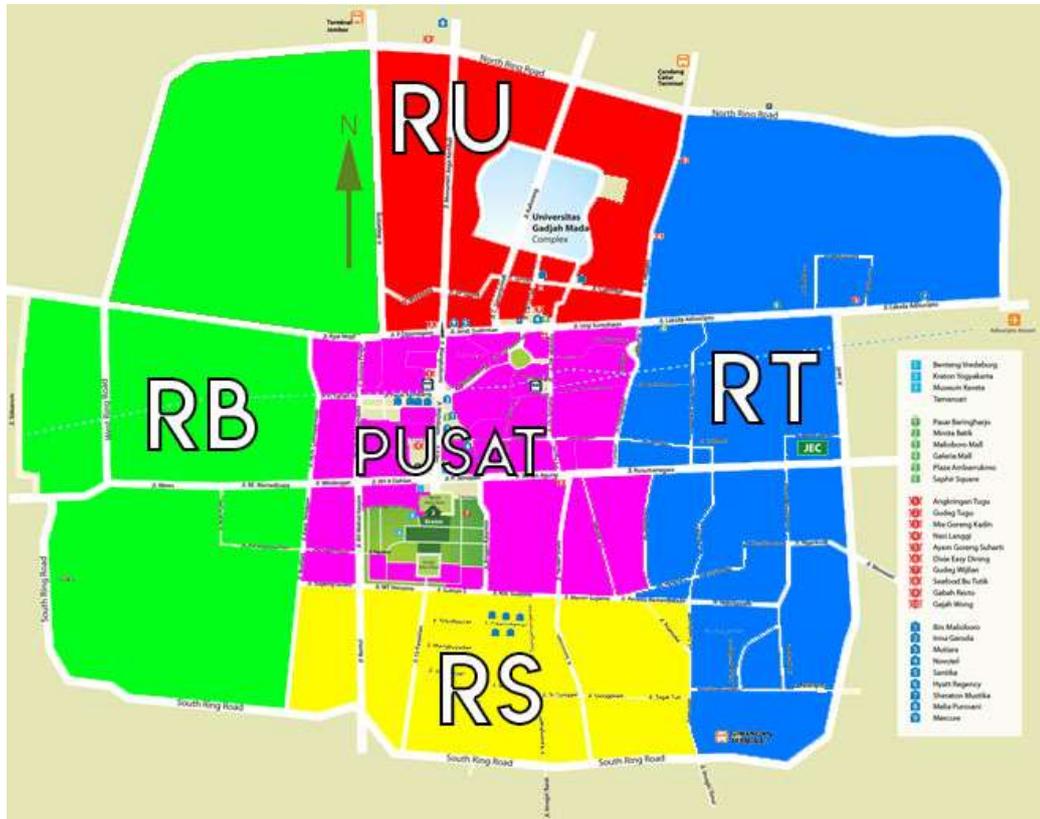
Data *input* adalah beberapa data yang penulis gunakan untuk membuat halaman yang nantinya akan digunakan untuk proses *input* maupun *edit* data. Data tersebut adalah data yang memiliki hubungan erat dengan pekerjaan admin perusahaan, adapun data tersebut sebagai berikut:

1. Nama Pemesan
2. Nomer Handphone Pemesan
3. Tanggal Pesan
4. Tanggal Antar
5. Alamat Antar
6. Jam Antar
7. Jenis Produk
8. Harga Jual
9. Jumlah Porsi
10. Ongkos Kirim
11. Total Harga
12. Uang Muka
13. Sisa Pembayaran

Adapun pada data ongkos kirim tersebut diisi berdasarkan daftar harga yang telah dibuat berdasarkan keputusan oleh pemilik usaha, dan berikut data tersebut dapat dilihat pada tabel 3.9.

Tabel 3.9 Daftar Biaya Ongkos Kirim

Dalam Kota					Luar Kota
R. Selatan	R. Barat	R. Utara	R. Timur	Pusat	
Rp. 25.000	Rp. 20.000	Rp. 35.000	Rp. 35.000	Rp. 25.000	JNE / POS / J&T



Gambar 3.29 Peta Pembagian Wilayah Ongkos Kirim

Pada gambar ini berisi tentang penjelasan peta pembagian wilayah ongkos kirim untuk daerah Yogyakarta. Adapun penjelasannya sebagai berikut:

1. Peta berwarna hijau adalah wilayah daerah sekitar jalan ringroad barat.
2. Peta berwarna merah adalah wilayah daerah sekitar jalan ringroad utara.
3. Peta ungu adalah wilayah daerah pusat kota jogja.
4. Peta berwarna biru adalah wilayah daerah sekitar ringroad timur.
5. Peta berwarna kuning adalah wilayah daerah sekitar ringroad selatan.