

## **BAB III**

### **METODOLOGI PENELITIAN**

#### **3.1 Metode Pengembangan Perangkat Lunak**

##### **3.1.1 Komunikasi**

Dalam metode *prototype*, langkah pertama yang dilakukan adalah komunikasi. Komunikasi adalah cara atau teknik yang dilakukan dalam memperoleh data pendukung penelitian. Teknik yang digunakan adalah sebagai berikut :

##### **a. Penelitian Kepustakaan (*Library Research*)**

Studi kepustakaan yang dilakukan oleh peneliti meliputi mempelajari jurnal penelitian yang berhubungan dengan *E-service*. Selain itu peneliti juga melakukan pengambilan data-data yang dibutuhkan untuk membuat konten pada website *E-service* pada *sherly snack* berupa data yang ada di *sherly snack*.

##### **b. Wawancara**

Peneliti juga melakukan pengambilan data dengan teknik wawancara. Pada tahap wawancara ini peneliti mengajukan beberapa pertanyaan kepada narasumber terkait hal-hal yang dibutuhkan oleh user. Tahapan wawancara ini narasumber yang diminta untuk menjawab pertanyaan-pertanyaan tersebut adalah admin *sherly snack*.

Setelah melakukan pengumpulan data, peneliti menganalisa sistem dari data yang telah dikumpulkan. Terlebih dahulu peneliti menganalisa terhadap prosedur yang berjalan, kemudian peneliti menganalisis kebutuhan sistem dan selanjutnya membangun *prototype* dan memperbaiki sistem *prototype* setelah itu menguji coba *prototype* yang dibangun sesuai dengan keinginan pembangun sistem. Apabila sistem tersebut dirasa kurang maka akan kembali ke tahap awal dari mengumpulkan data kembali dan menjalankan proses selanjutnya.

##### **c. Dokumentasi**

Dokumentasi merupakan catatan peristiwa yang sudah berlalu. dokumen bisa berbentuk tulisan, gambar, atau karya-karya monumental dari seseorang. pada

penelitian ini, metode ini dapat dilakukan dengan cara mengumpulkan data barang yang ada di sherly snack.

### **3.1.2 Perencanaan Secara Cepat**

#### **3.1.2.1 Analisis Kebutuhan Pengguna**

Pada tahap ini, peneliti berkerja sama dengan pengguna dari sistem yang akan diusulkan untuk memperoleh informasi dasar yang dibutuhkan oleh pengguna terhadap sistem. Pengguna dari sistem ini dibagi menjadi 3 :

- 1) Admin : orang yang di izinkan untuk mengawasi dan mengontrol akses informasi dan data yang ada pada website tersebut.
- 2) Pelanggan : pelanggan dari sherly snack
- 3) *Supplier* : adalah pemasok barang yang ada di sherly snack

Berdasarkan analisa kebutuhan pengguna diperoleh beberapa informasi yang diperoleh diantaranya :

- 1) Perlunya sebuah media informasi yang dapat menambah Informasi kepada pelanggan.
- 2) Perlu adanya sebuah media yang dapat mempermudah pelanggan dalam melakukan pengecekan barang yang dibutuhkan sehingga tidak perlu harus datang ke sherly snack.
- 3) Adanya fasilitas transaksi yang mudah untuk mempermudah transaksi pelanggan.
- 4) Menyediakan fasilitas kepada *supplier* untuk menambahkan produk, menghapus produk, mengubah produk.

#### **3.1.2.2 Analisis Perangkat Lunak**

Analisis kebutuhan perangkat lunak yang digunakan untuk membangun sebuah sistem *website* adalah sebagai berikut :

- a. Perangkat lunak sistem operasi yang digunakan adalah *Microsoft Windows 7 64Bit*.
- b. Perangkat lunak aplikasi yang digunakan
  - a. *Web server* menggunakan *xampp*.
  - b. *Database* menggunakan *MySQL*.
  - c. *Editor Interface* menggunakan *sublime text 3*
  - d. *Editor gambar* menggunakan *Adobe Photoshop CS6*.
  - e. *Visio* untuk mendesain rancangan UML.

### 3.1.2.3 Analisis Perangkat Keras

Adapun spesifikasi perangkat keras yang digunakan untuk membangun sebuah sistem tersebut adalah sebagai berikut :

1. *Processor Core i3*.
2. *Monitor 14"*.
3. *Ram 4 Gb*.
4. *Harddisk 500 GB*.
5. *Keyboard dan mouse optik*.

### 3.1.2.4 String Matching

*String Matching* adalah proses pencarian semua kemunculan *query* yang selanjutnya disebut *pattern* ke dalam *string* yang lebih panjang atau teks (Aan et al., 2017). *String Matching* dirumuskan sebagai berikut :

$$x = x[0 \dots m-1] \quad y = y[0 \dots n-1]$$

Dimana :

*x* adalah *pattern*

*m* adalah panjang *pattern*

*y* adalah teks

*n* adalah panjang teks

Kedua *string* terdiri dari sekumpulan karakter yang disebut alfabet yang

dilambangkan dengan  $\Sigma$  (sigma) dan mempunyai ukuran  $\sigma$  (tao). *String matching* dibagi menjadi dua, yakni *exact matching* dan *heuristic* atau *statistical matching*. *Exact Matching* digunakan untuk menemukan pattern yang berasal dari satu teks. Contoh pencarian *exact matching* adalah pencarian kata “pelajar” dalam kalimat “saya seorang pelajar” atau “saya seorang siswa”. Sistem akan memberikan hasil bahwa kalimat pertama mengandung kata “pelajar” sedangkan kalimat kedua tidak, meskipun kenyataannya pelajar dan siswa adalah kata yang bersinonim. Algoritma *exact matching* diklasifikasi menjadi tiga bagian menurut arah pencariannya, yaitu :

1. Arah pembacaan dari kiri ke kanan.

Algoritma yang termasuk kategori ini adalah *Brute Force*, Morris dan Pratt (yang kemudian dikembangkan oleh Knuth, Morris, dan Pratt).

2. Arah pembacaan dari kanan ke kiri.

Algoritma yang termasuk kategori ini adalah *Boyer* dan *Moore* yang kemudian dikembangkan menjadi algoritma *turbo Boyer-Moore*, *tuned Boyer-Moore*, dan *Zhu-Takaoka*.

3. Arah pencarian yang ditentukan program. Algoritma yang termasuk kategori ini adalah algoritma *Colussi*, *Crochemore-Perrin*.

*Heuristic matching* adalah teknik yang digunakan untuk menghubungkan dua data terpisah ketika *exact matching* tidak mampu mengatasi karena pembatasan pada data yang tersedia (Aan et al., 2017). *Heuristic matching* dapat dilakukan dengan perhitungan *distance* antara *pattern* dengan teks. *Exact* dan *heuristic matching* memiliki kelemahan dalam menemukan kata yang memiliki kemiripan makna tetapi berbeda tulisan.

### **3.1.2.5 Algoritma Brute Force**

Algoritma *brute force* adalah algoritma untuk mencocokkan *pattern* dengan semua teks antara 0 dan n-m untuk menemukan keberadaan *pattern* dalam teks (Aan et al., 2017) Di dalam pencocokkan string, terdapat istilah teks dan *pattern*. Teks merupakan kata yang dicari dan dicocokkan dengan *pattern*. Sedangkan *pattern*

merupakan kata yang di inputkan untuk dicocokkan. Secara rinci, langkah–langkah yang dilakukan algoritma ini saat mencocokkan string adalah:

1. Algoritma *brute force* mulai mencocokkan *pattern* dari awal teks.
2. Dari kiri ke kanan, algoritma ini akan mencocokkan karakter per karakter *pattern* dengan karakter pada teks yang bersesuaian, sampai salah satu kondisi berikut terpenuhi:
  - a. Karakter di *pattern* dan di teks yang dibandingkan tidak cocok.
  - b. Semua karakter di *pattern* cocok. Kemudian algoritma akan memberitahukan penemuan di posisi ini.
3. Algoritma kemudian terus menggeser *pattern* sebesar satu ke kanan, dan mengulangi langkah ke-2 sampai *pattern* berada di ujung teks.

Algoritma *brute force* juga memiliki kelebihan dan kelemahan. Adapun kelebihan dari algoritma *brute force* yaitu :

1. Algoritma *brute force* dapat digunakan untuk memecahkan hampir sebagian besar masalah.
2. Algoritma *brute force* sederhana dan mudah dimengerti.
3. Algoritma *brute force* menghasilkan algoritma yang layak untuk beberapa masalah penting seperti pencarian, pengurutan, pencocokkan string, atau perkalian matriks.
4. Algoritma *brute force* menghasilkan algoritma baku (standart) untuk tugas-tugas komputasi penjumlahan/perkalian n buah bilangan, menentukan elemen minimum atau maksimum di dalam tabel (list).

Sedangkan kelemahan dari algoritma *brute force* yaitu sebagai berikut :

1. Algoritma *brute force* jarang menghasilkan algoritma yang mangkus (manjur).
2. Beberapa algoritma *brute force* lambat, sehingga tidak dapat diterima.
3. Tidak sekonstruktif/sekreatif teknik pemecahan masalah lainnya.

Contoh penggunaan algoritma *Brute Force* untuk pencarian *pattern* dalam teks :

Teks = WAFER COKLAT

*Pattern* = COKLAT

**Tabel 3.1** langkah ke I

Langkah Ke I												
Text	W	A	F	E	R		C	O	K	L	A	T
Patter	C	O	K	L	A	T						
Indeks	0	1	2	3	4	5	6	7	8	9	10	11

geser *pattern* sebanyak satu langkah ke kanan menuju indeks berikutnya

**Tabel 3.2** Langkah Ke II

Langkah Ke II												
Text	W	A	F	E	R		C	O	K	L	A	T
Patter		C	O	K	L	A	T					
Indeks	0	1	2	3	4	5	6	7	8	9	10	11

geser *pattern* sebanyak satu langkah ke kanan menuju indeks berikutnya

**Tabel 3.3** Langkah Ke III

Langkah Ke III												
Text	W	A	F	E	R		C	O	K	L	A	T
Patter			C	O	K	L	A	T				
Indeks	0	1	2	3	4	5	6	7	8	9	10	11

geser *pattern* sebanyak satu langkah ke kanan menuju indeks berikutnya

**Tabel 3.4** langkah ke 1V

Langkah Ke IV												
Text	W	A	F	E	R		C	O	K	L	A	T
Patter				C	O	K	L	A	T			
Indeks	0	1	2	3	4	5	6	7	8	9	10	11

geser *pattern* sebanyak satu langkah ke kanan menuju indeks berikutnya

**Tabel 3.5** Langkah Ke V

Langkah Ke V												
Text	W	A	F	E	R		C	O	K	L	A	T
Patter					C	O	K	L	A	T		

Indeks	0	1	2	3	4	5	6	7	8	9	10	11
--------	---	---	---	---	---	---	---	---	---	---	----	----

geser *pattern* sebanyak satu langkah ke kanan menuju indeks berikutnya

**Tabel 3.6** Langkah Ke VI

Langkah Ke VI												
Text	W	A	F	E	R		C	O	K	L	A	T
Patter							C	O	K	L	A	T
Indeks	0	1	2	3	4	5	6	7	8	9	10	11

geser *pattern* sebanyak satu langkah ke kanan menuju indeks berikutnya

**Tabel 3.7** Langkah Ke VII

Langkah Ke VII												
Text	W	A	F	E	R		C	O	K	L	A	T
Patter							C	O	K	L	A	T
Indeks	0	1	2	3	4	5	6	7	8	9	10	11

geser *pattern* sebanyak satu langkah ke kanan menuju indeks

Dari langkah algoritma brute force dalam mencari patern dalam teks diatas algoritma menelusuri satu persatu teks dengan petern bergeser satu langkah ke kanan dengan menentukan kecocokan teks yang dicari ketika teks dan patern bernilai sama maka akan berhenti dan menampilkan kata yang dicari seperti penjelasan tabel-tabel diatas yaitu pada teks wafer coklat dan petern coklat algoritma akan mencari kecocokan string dengan kecocokan yang bernilai sama di teks lalu algoritma menemukan teks yan sesuai dengan pattern yang dicari maka patern akan berhenti di teks yang dicari.

### 3.1.2.6 Framework Dan Bahasa Pemrograman Yang Digunakan

*Framework* dan Bahasa pemrograman yang digunakan akan dijelaskan berikut ini :

#### a. *Framework Bootstraps*

Menurut (Alatas, 2013) *bootstrap* merupakan *framework* ataupun *tools* untuk membuat aplikasi *website* ataupun situs *web responsive* secara cepat, mudah dan gratis, karena *website* yang dibangun oleh peneliti merupakan *website* yang dapat

diakses dalam perangkat *mobile* ataupun *personal computer*. *Bootstrap* terdiri dari *css* dan *html* untuk menghasilkan *grid, layout, typography, table, form, navigation*, dan lain lain. Di dalam *bootstrap* juga sudah terdapat *jquery plugins* untuk menghasilkan komponen ui yang cantik seperti *transitions, modal, dropdown, scrollspy, tooltip, tab, popever, alert, button, carousel*, dan lain lain.

### **b. Bahasa Pemrograman PHP**

PHP singkatan dari *Hypertext Preprocessor* yang merupakan *server-side programming*, yaitu bahasa pemrograman yang diproses di sisi server. Fungsi utama PHP dalam membangun website adalah untuk melakukan pengolahan data pada database. Data website akan dimasukkan ke database, diedit, dihapus dan ditampilkan pada website yang diatur oleh PHP.(Abdulloh, 2016)

### **c. Bahasa Pemrograman Javascript**

Berbeda dengan php yang diproses di sisi server, javascript diproses pada komputer client. Karena pemrosesannya dilakukan di komputer client, membuat javascript lebih interaktif dibanding php. Peran javascript dalam membuat website adalah memberikan efek animasi yang menarik dan interaktivitas dalam penanganan event yang dilakukan oleh pengguna website.(Abdulloh, 2016)

### **3.1.2.7 Unsur Website**

unsur website yang digunakan untuk kebutuhan perancangan akan dijelaskan berikut ini :

#### **a. Nama Domain**

Pengertian nama domain atau biasa disebut dengan Domain Name atau URL adalah alamat unik di dunia internet yang digunakan untuk mengidentifikasi sebuah website, atau dengan kata lain domain name adalah alamat yang digunakan untuk menemukan sebuah website pada dunia internet.(Utama, 2011)



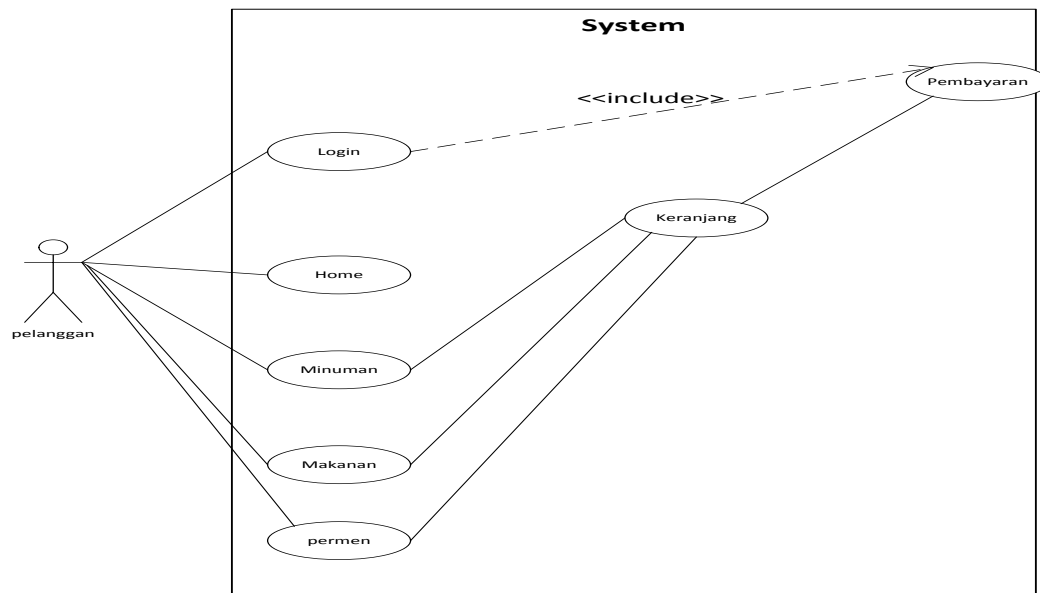
## b. Rumah Tempat Website (Web Hosting)

Pengertian Web Hosting dapat diartikan sebagai ruangan yang terdapat dalam harddisk tempat menyimpan berbagai data, file, gambar dan lain sebagainya yang akan ditampilkan di website. Besarnya data yang bisa dimasukkan tergantung dari besarnya web hosting semakin besar pula data yang dapat dimasukkan dan ditampilkan dalam website. Web Hosting juga juga diperoleh dengan menyewa besarnya hosting ditentukan ruangan harddisk dengan ukuran MB (Mega Byte) atau GB (Giga Byte). Lama penyewaan web hosting rata rata dihitung per tahun. Penyewaan hosting dilakukan dari perusahaan perusahaan penyewa web hosting yang banyak dijumpai baik di Indonesia maupun luar negeri.(Utama, 2011)

### 3.1.3 Pemodelan Perencanaan Secara Cepat

#### 3.1.3.1 Analisis Pelanggan yang Diusulkan Pada Sistem Aplikasi *E-Service*

Penulis mengusulkan merancang sistem yang dapat diakses oleh user melalui komputer pribadi maupun melalui *smartphone*. Untuk melihat rancangan yang diusulkan pada user dapat dilihat pada gambar 3.1 dibawah ini.



**Gambar 3.1** Use case Pelanggan pada sistem yang diusulkan

### 3.1.3.1.1 Definisi Aktor *Use Case* Sistem Untuk Pelanggan Yang Diusulkan

Definisi aktor pelanggan merupakan penjelasan dari apa yang dilakukan oleh aktor yang terlibat dalam perangkat lunak yang dibangun. Adapun definisi aktor Pelanggan pada sistem yang diusulkan dapat dilihat pada tabel 3.8 berikut ini.

**Tabel 3.8** Penjelasan Aktor Sistem Untuk Pelanggan Pada Aplikasi *E-service* yang diusulkan.

Aktor	Deskripsi
Pelanggan	1. Pelanggan Mengakses Sistem 2. Pelanggan dapat melakukan beberapa pilihan menu diantaranya menu home, makanan, minuman. 3. Pelanggan ketika mengakses menu makanan dan minuman dapat melakukan memilih barang yang akan ditampung dikeranjang akantetapi pelanggan harus masuk terlebih dahulu sebelum menyelesaikan pembayaran.

### 3.1.3.1.2 Analisa *Use Case* Yang Diusulkan Pada *User*

Aktor : pelanggan.

Tujuan : untuk mengakses pasilitas layana yang ada di *e-service*.

Deskripsi : pelanggan dapat menggunakan pasilitas yang ada di *e-service* diantaranya memilih produk dan melihat produk-produk yang dibutuhkan yang akan dijelaskan pada tabel 3.9 berikut ini.

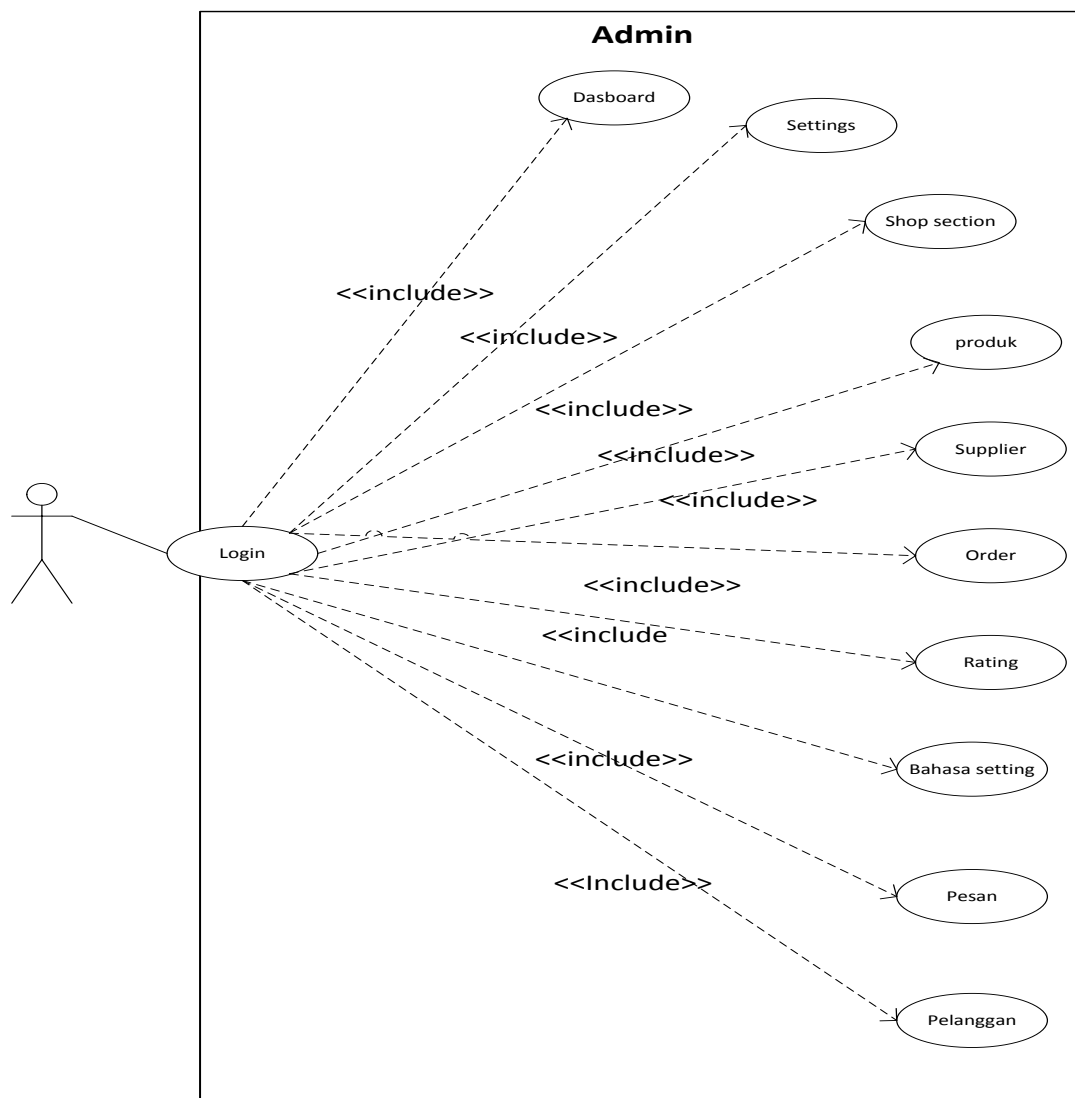
**Tabel 3.9** penjelasan *usecase user* yang diusulkan

USER	SISTEM
1. Mengakses Sistem	2. Menampilkan halaman menu utama
3. Memanfaatkan fasilitas menu	4. Menampilkan menu yang dipilih
5. User Memilih Barang	6. Menampung barang yang dipilih

	dikeranjang
7. Melakukan pembayaran	8. Memberi informasi bahwa anda harus login

### 3.1.3.2 Analisis Admin yang Diusulkan Pada Sistem Aplikasi *E-Service*

Analisis sistem admin yang dirancang mulai dari admin melakukan login memanfaatkan menu yang akan dijelaskan dengan gambar 3.2 dibawah ini.



**Gambar 3.2** Use case Admin yang diusulkan

### 3.1.3.2.1 Definisi Aktor *Use Case* Sistem Untuk Admin Yang Diusulkan

Definisi aktor Admin merupakan penjelasan dari apa yang dilakukan oleh aktor yang terlibat dalam perangkat lunak yang dibangun. Adapun definisi aktor admin pada sistem yang diusulkan dapat dilihat pada tabel 3.10 dibawah ini.

**Tabel 3.10** Penjelasan Aktor Sistem Untuk Admin Pada Aplikasi *E-service* yang diusulkan.

Aktor	Deskripsi
Admin	1. Admin mengakses sistem 2. Admin memanfaatkan fungsi <i>tolls</i> yang ada 3. Setelah admin berhasil masuk ke sistem admin dapat melakukan <i>seting</i> , <i>ubdate</i> dan melihat pesanan pelanggan

### 3.1.3.2.2 Analisa *Use case* Admin Yang diusulkan

Aktor : admin.

Tujuan : untuk mengontrol seluruh fasilitas layana yang ada di *e-service*.

Deskripsi : admin dapat menggunakan fasilitas yang ada di *e-service* yang akan dijelaskan pada tabel 3.11 dibawah ini.

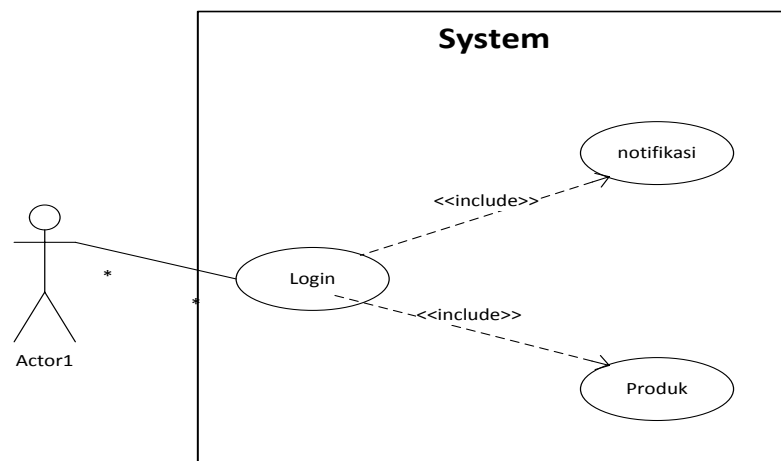
**Tabel 3.11** Analisa *Usecase* Admin

AKTOR	SISTEM
1. Admin mengakses halaman admin	
	2. Sistem akan menampilkan form <i>login</i>
3. Admin mengisi <i>username</i> dan <i>password</i>	
	4. Sistem akan memeriksa apakah data admin ada atau

	tidak
	5. Jika tidak ada maka kembali ke point 3
	6. jika ada maka masuk ke halaman utama admin
7. Admin dapat melihat informasi dan memanipulasi data seperti <i>insert</i> , <i>update</i> , <i>read</i> dan <i>delete</i> .	
	8. Data akan disimpan ke <i>database</i> .
9. Admin <i>logout</i> atau keluar dari halaman utama admin.	

### 3.1.3.3 Analisis *Supplier* yang diusulkan pada *E-service*

Analisis sistem *supplier* yang dirancang mulai dari *supplier* melakukan login melihat notifikasi jika ada stok habis, melakukan update produk sampai *supplier logout* kembali. Untuk lebih lengkapnya penjelasan tersebut dapat dilihat pada gambar 3.3 dibawah ini.



**Gambar 3.3** *Usecase Supplier*

### 3.1.3.3.1 Definisi Aktor *Use Case* Sistem Untuk *Supplier* Yang Diusulkan

Definisi aktor *supplier* merupakan penjelasan dari apa yang di lakukan oleh aktor yang terlibat dalam perangkat lunak yang di bangun. Adapun definisi aktor *supplier* pada sistem yang diusulkan dapat dilihat pada tabel 3.12 dibawah ini.

**Tabel 3.12** Penjelasan Aktor Sistem Untuk *Supplier* Pada Aplikasi *E-Service* yang Diusulkan.

Aktor	Deskripsi
Admin	<ol style="list-style-type: none"> <li>1. <i>Supplier</i> mengakses sistem</li> <li>2. <i>Supplier</i> memanfaatkan fungsi tols yang ada</li> <li>3. Setelah <i>Supplier</i> berhasil masuk ke sistem. <i>supplier</i> dapat melakukan input produk, edit produk, hapus produk dan melihat notifikasi barang yang habis.</li> </ol>

### 3.1.3.3.2 Analisis *Supplier* yang diusulkan

Aktor : *Supplier*

Tujuan : untuk melihat stok dan melakukan update produk

Deskripsi : dapat Menggunakan Fasilitas yang ada di *e-service* yang akan dijelaskan pada table 3.13 dibawah ini.

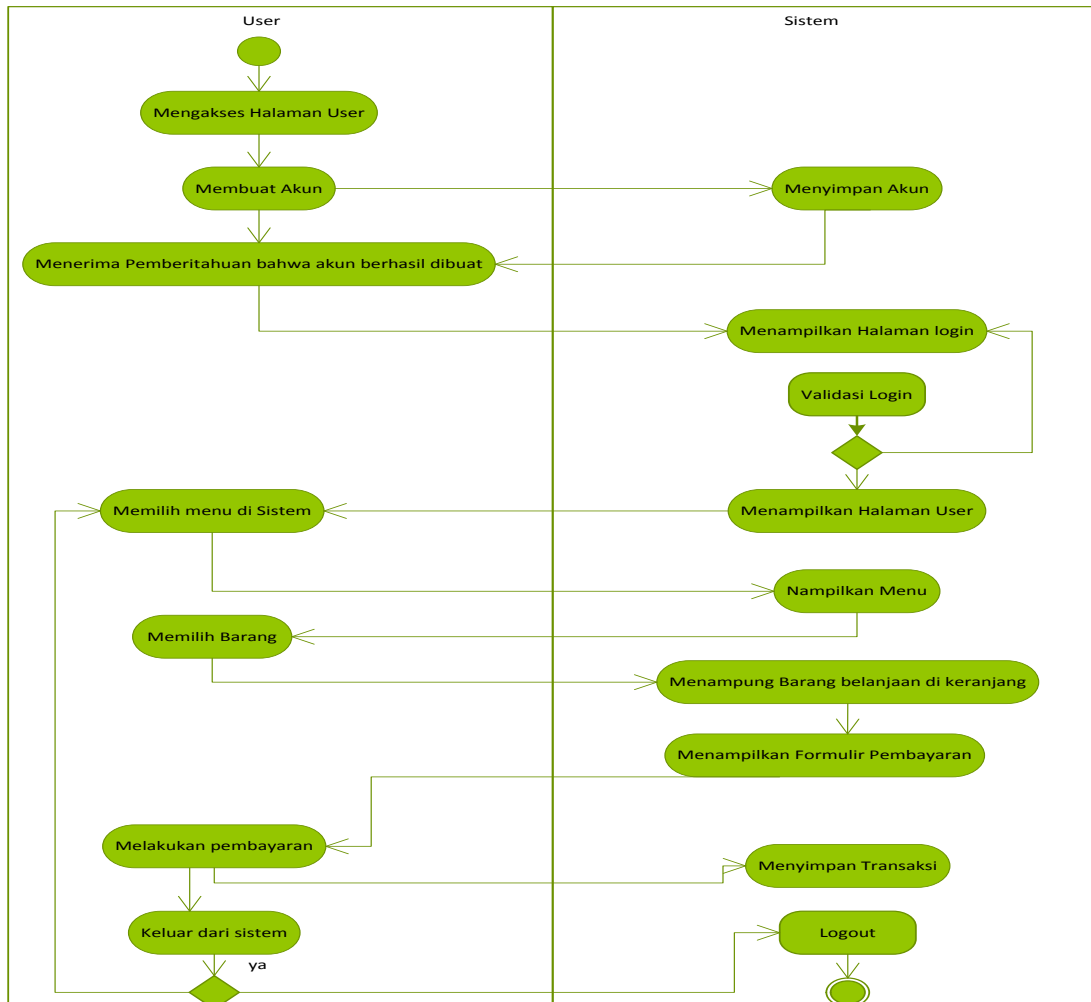
Table 3.13 Analisa *Usecase* *Supplier*

<i>Supplier</i>	SISTEM
1. <i>Supplier</i> mengakses halaman <i>supplier</i>	
	2. Sistem akan menampilkan form login
3. <i>Supplier</i> mengisi <i>username</i> dan <i>password</i>	
	4. Sistem akan memeriksa

	apakah data <i>Supplier</i> ada atau tidak
	5. Jika tidak ada maka kembali ke point 3
	6. jika ada maka masuk ke halaman utama <i>Supplier</i>
7. <i>Supplier</i> dapat melihat Notifikasi dan melakukan <i>update</i> stok prodak.	
	8. Data akan disimpan ke database.
9. <i>Supplier</i> logout atau keluar dari halaman utama <i>Supplier</i> .	

#### 3.1.3.4 Activity Diagram Pelanggan yang Diusulkan Pada Sistem *E-service*

*Activiti diagram* user yang diusulkan yaitu mulai dari *user* mengases sistem setelah melakukan pendaftaran maka sistem menyimpan akun dan *user* menerima informasi bahwa akun berhasil dibuat dan sistem menampilkan halaman login dan sistem melakukan validasi login jika sukses maka *user* diarahkan ke halaman menu setelah itu *user* memilih barang dan sistem menyimpan barang yang dipilih di keranjang setelah itu sistem mengarahkan *user* ke form pembayaran setelah melakukan pembayaran dan sistem menyimpan di database lalu *user* mendapat informasi bahwa transaksi sukses dapat dilihat pada gambar 3.4 dibawah ini.

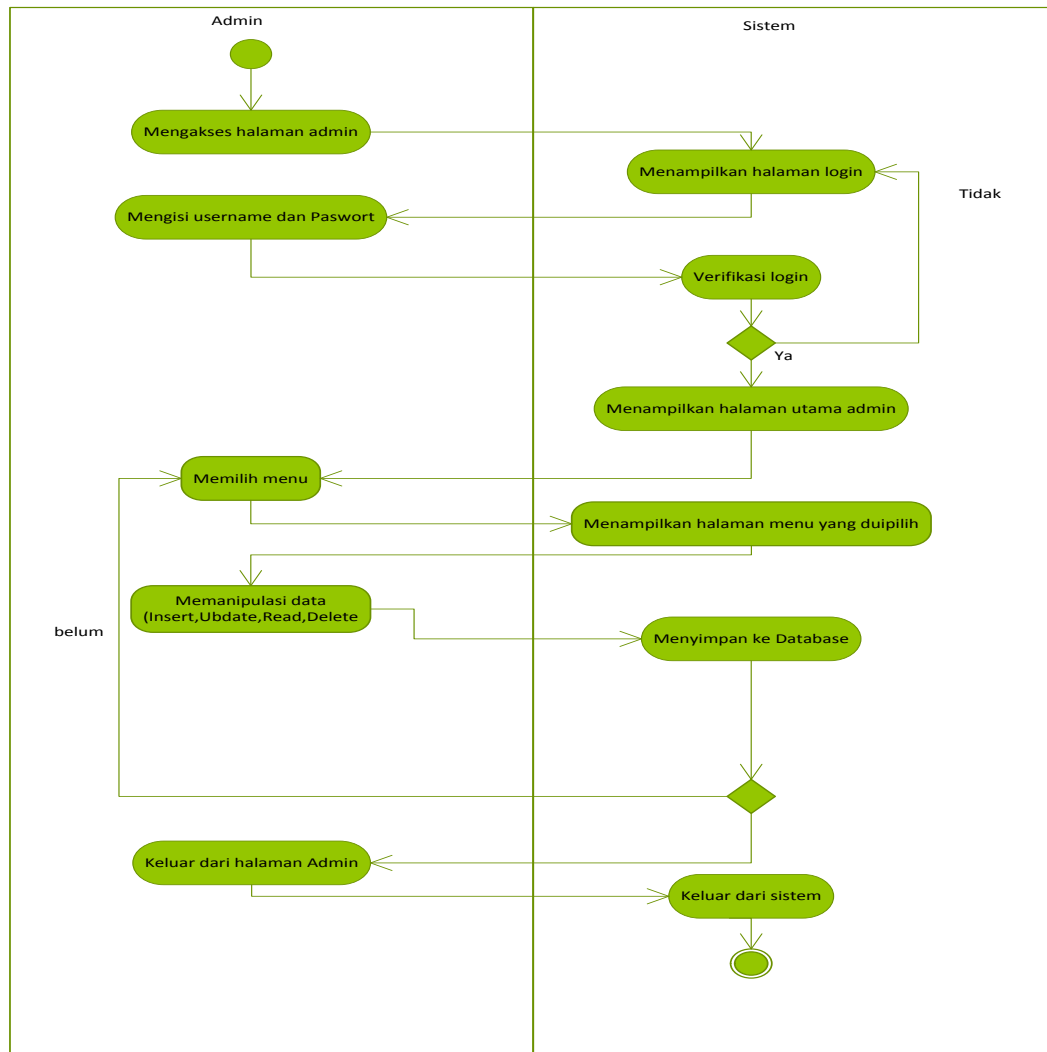


**Gambar 3.4** Activity Diagram Pelanggan

### 3.1.3.5 Activity Diagram Admin Yang Diusulkan Pada Sistem E-Service

Adapun *activity* diagram admin yang diusulkan pada *e-service* yaitu mulai dari admin mengakses halaman login sistem menampilkan halaman login setelah itu admin mengisi form login dan sistem menampilkan menu jika gagal maka sistem mengarahkan untuk login, setelah itu admin memilih menu dan sistem menampilkan halaman menu yang dipilih admin memanipulasi data (*insert,update,read,delete*) lalu sistem menyimpan perintah admin ke database lalu admin *logout*, jika belum maka admin tetap berada di halaman menu dapat dilihat pada gambar 3.5 dibawah ini.



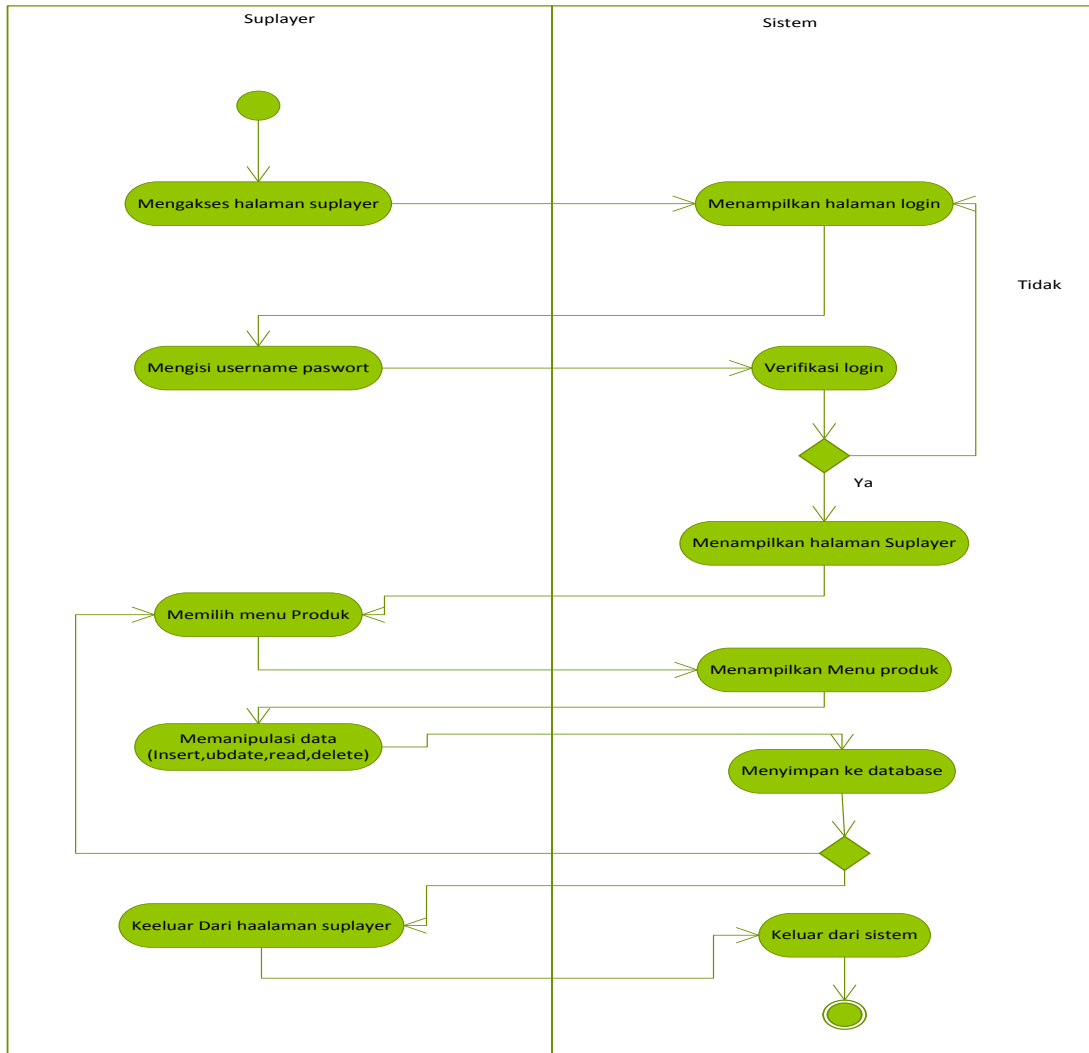


**Gambar 3.5** Activity Diagram Admin

### 3.1.3.6 Activity Diagram Supplier yang Diusulkan Pada Sistem

Activity diagram *supplier* yang diusulkan pada sistem yaitu *supplier* mengakses halaman *supplier* dan sistem menampilkan halaman login *supplier* mengisi form login dan *supplier* melakukan *verifikasi* login jika sukses maka akan ke menu utama *supplier* namun jika gagal akan kembali ke halaman utama login setelah berhasil login maka *supplier* akan melihat pesan setelah itu membaca dan mengirim pesan lalu

sistem menyimpan ke database lalu *supplier* keluar dari halaman *supplier* dapat dilihat pada gambar 3.6 dibawah ini.

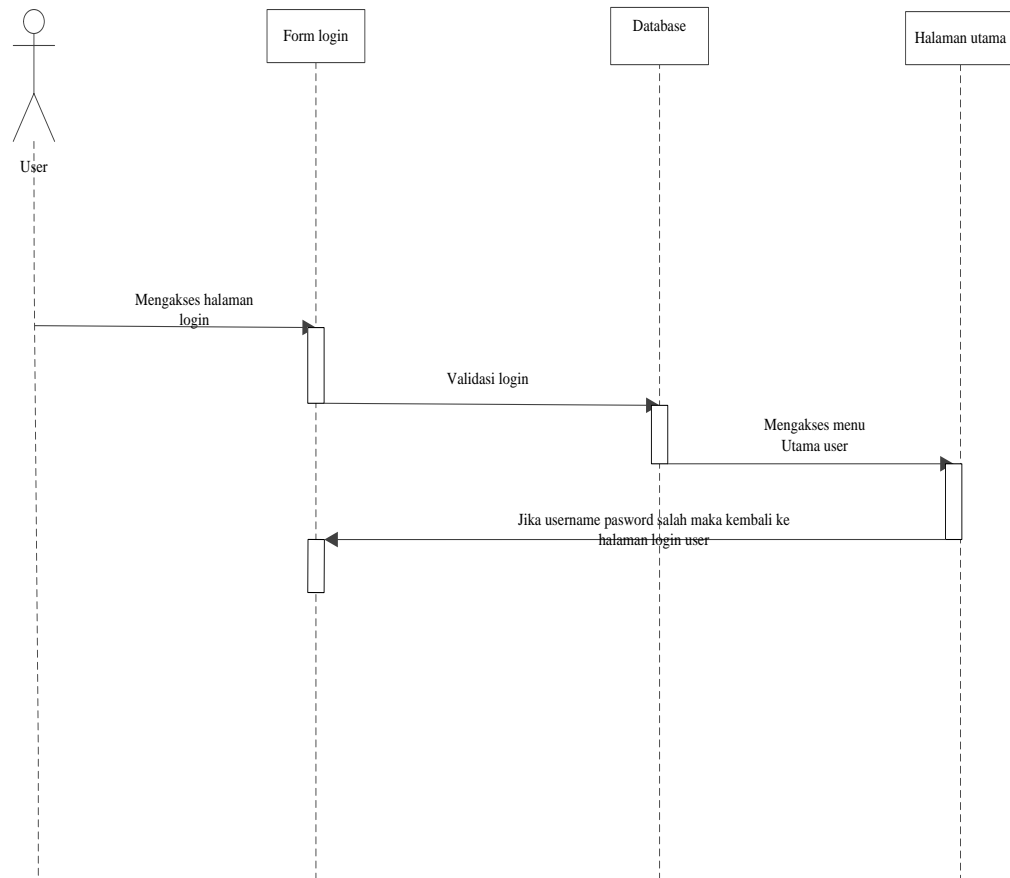


**Gambar 3.6** Activity Diagram Supplier

### 3.1.3.7 Sequence Diagram Pelanggan Untuk Sistem Login yang Diusulkan

Pada *sequence* diagram menerangkan serangkaian hubungan yang terjadi antara *user* dan halaman utama login serta sistem *database*. Dalam diagram ini *user* mengakses halaman login, kemudian sistem akan melakukan *verifikasi* dari *database*, jika data tersebut ada maka akan masuk kehalaman utama *user*, jika tidak ada maka kembali

kehalaman utama login. *Sequence diagram* pelanggan untuk sistem login yang diusulkan dapat dilihat pada gambar 3.7 dibawah ini

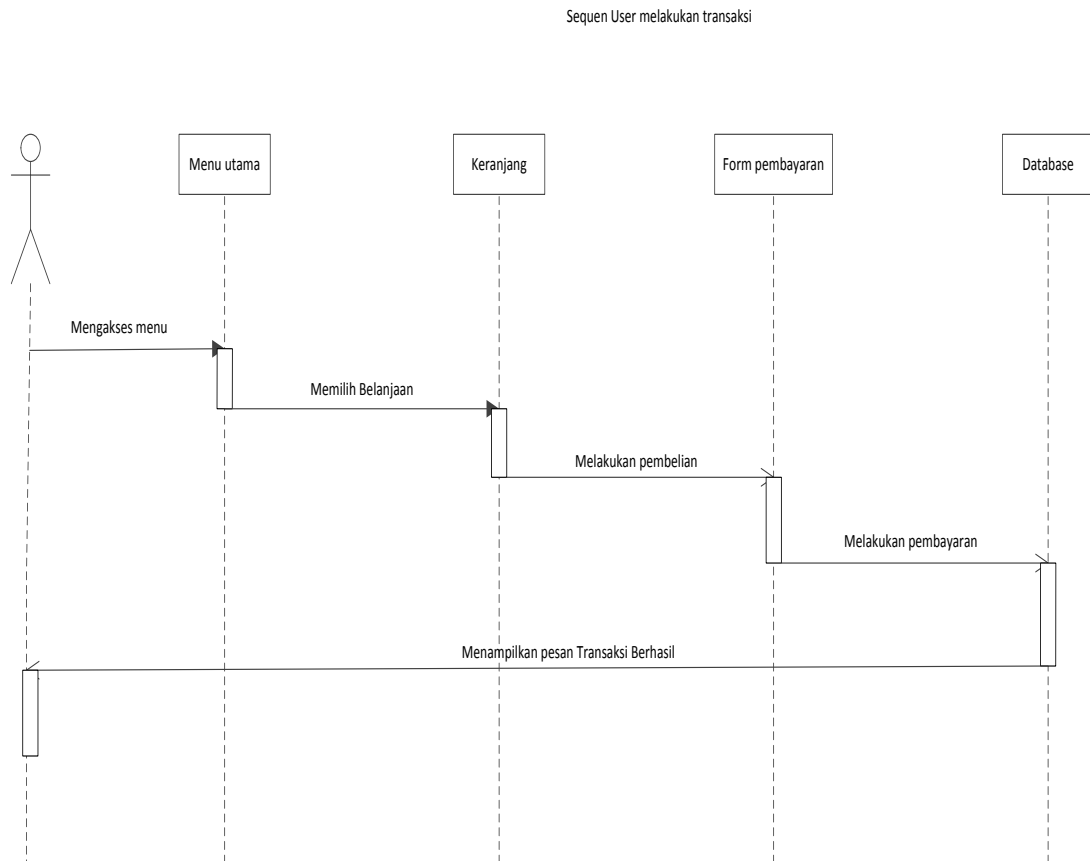


**Gambar 3.7** *sequence diagram* pelanggan untuk sistem login yang diusulkan

### 3.1.3.8 *Sequence* Pelanggan Melakukan Transaksi

Pada *sequence diagram* menerangkan serangkaian hubungan yang terjadi antara *user* dan halaman user melakukan transaksi serta sistem *database*. Dalam diagram ini *user* mengakses menu utama setelah itu user memilih belanjaan dan sistem menampung barang belanjaan di keranjang setelah itu sistem mengarahkan *user* ke menu *form* pembayaran setelah *user* melakukan *validasi* pembayaran sistem menyimpan

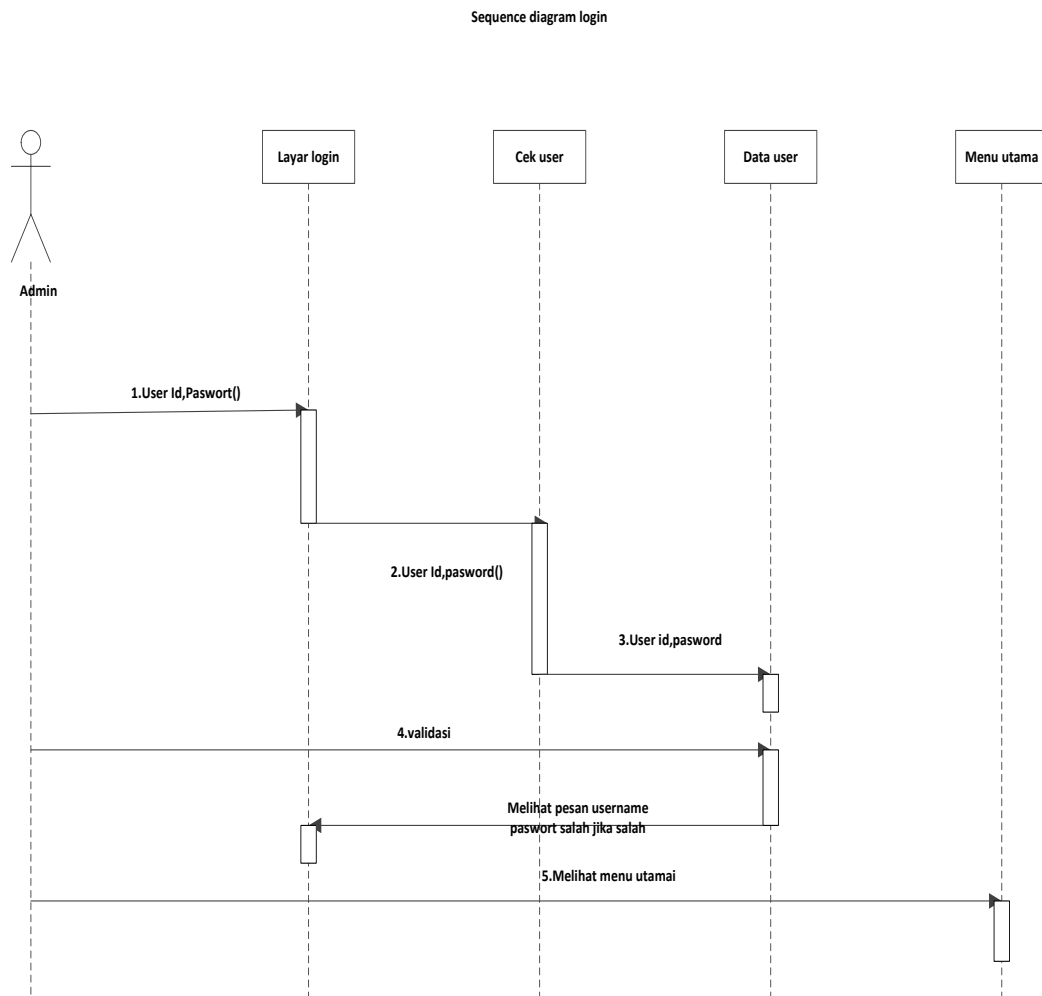
transaksi ke database dan memberi informasi ke *user* bahwa transaksi berhasil. *Sequence user* melakukan transaksi dapat dilihat pada gambar 3.8 dibawah ini.



**Gambar 3.8** *sequence user* melakukan transaksi

### 3.1.3.9 Sequence Diagram Admin Untuk Sistem Login Yang Diusulkan

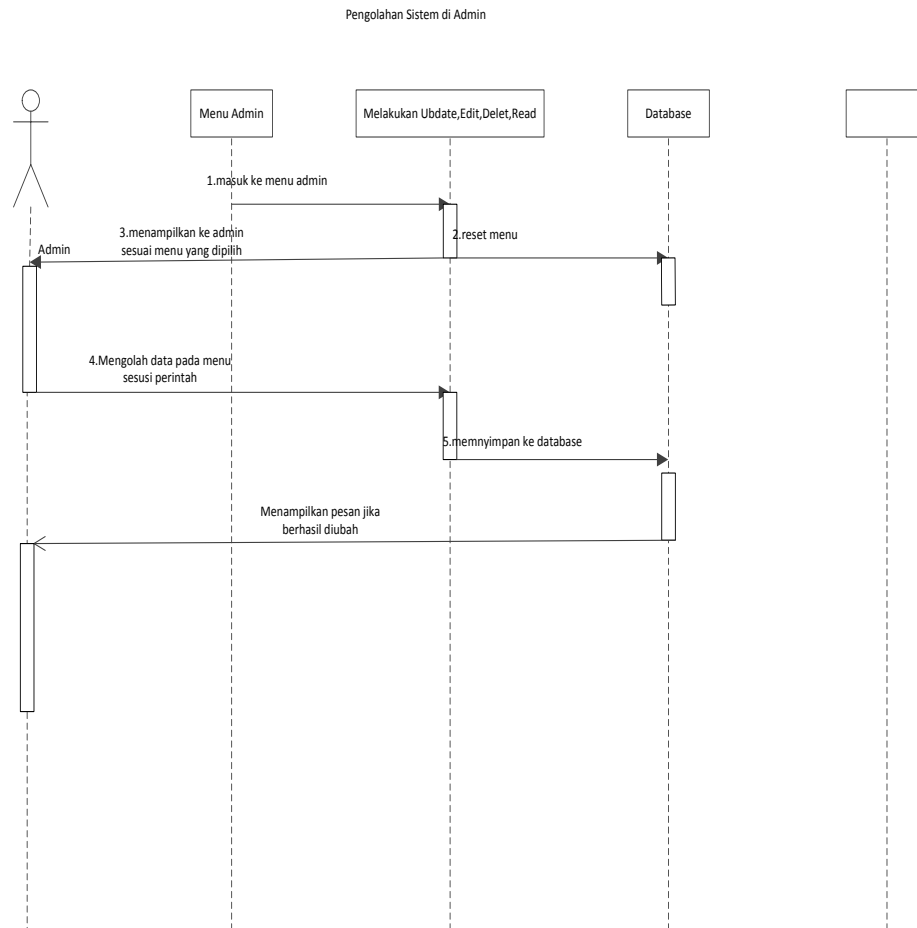
Pada *Sequence* diagram menerangkan serangkaian hubungan yang terjadi antara admin dan halaman utama login serta sistem *database*. Dalam diagram ini Admin mengakses halaman login, kemudian sistem akan melakukan *verifikasi* dari *database*, jika data tersebut ada maka akan masuk kehalaman utama Admin, jika tidak ada maka kembali kehalaman utama login. *Sequence diagram* admin untuk sistem login yang diusulkan dapat dilihat pada gambar 3.9 dibawah ini.



**Gambar 3.9** *sequence* admin login

### 3.1.3.10 *Sequence Diagram* Pengolahan Sistem Admin Yang Diusulkan

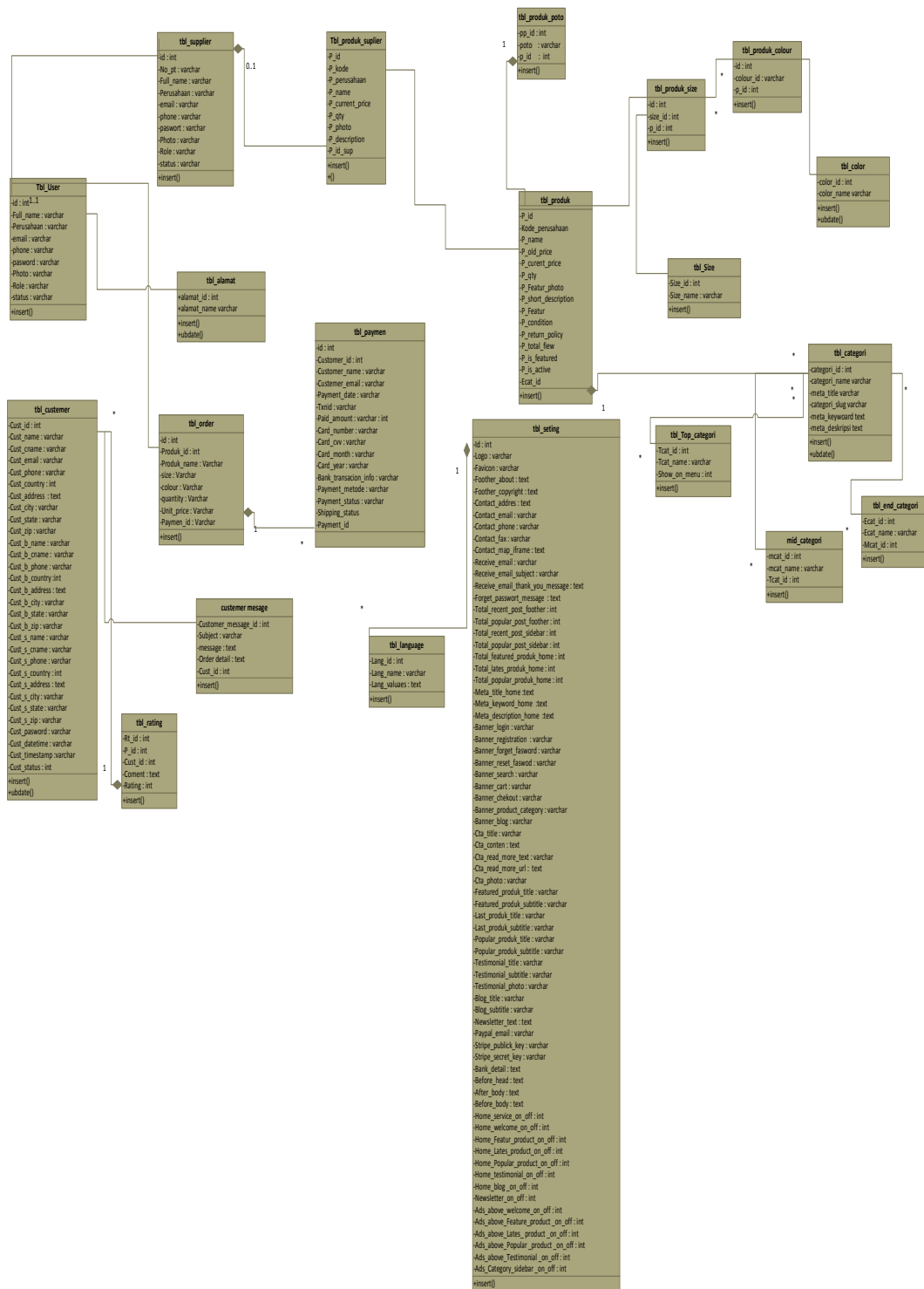
Pada pengolahan sistem di menu admin menjelaskan bahwa hubungan yang terjadi antara sistem admin dan database. *Sequence diagram* pengolahan sistem admin yang diusulkan dapat dilihat pada gambar 3.10 dibawah ini.



**Gambar 3.10** *sequence* pengolahan sistem admin

### 3.1.3.11 Class Diagram Sistem Aplikasi E-Service Terintegrasi

*Class diagram* merupakan alat bantu untuk menentukan langkah-langkah kerja yang akan dilakukan oleh pemrogram dimulai dari proses pengumpulan data, sampai pembentukan tabel sesuai dengan permasalahan yang ditangani. *Class diagram* ini terlebih dahulu dirancang dalam mendukung rencana pengolahan data elektronik supaya dapat berjalan dengan lebih baik, dan dengan relasi yang baik akan diperoleh gambaran umum sistem yang akan dipersiapkan dapat dilihat pada gambar 3.11 dibawah ini.



Gambar 3.11 Class diagram

### 3.1.3.12 Rancangan Struktur Database

Rancangan database yang digunakan untuk menyimpan data pada aplikasi E-service penjualan produk adalah Database Mysql dimana Struktur database yang akan dibuat dapat dilihat dibawah ini.

a. Struktur Tabel *Category*

Tabel ini digunakan untuk menyimpan data kategori.struktur tabel *category* dapat dilihat pada tabel 3.14 dibawah ini.

Nama database : db\_sherli

Nama tabel : *Category*

Primary key : *Category\_id*

Media penyimpanan : *Harddisk*

**Tabel 3.14** Struktur Tabel *Category*

No	Nama <i>Field</i>	<i>Type</i>	<i>Size</i>	Keterangan
1	<i>Category_id</i>	<i>Int</i>	11	Id kategori
2	<i>Category_name</i>	<i>Varchar</i>	255	Nama kategori
3	<i>Category_slug</i>	<i>Varchar</i>	100	Kategori slug
4	<i>Meta_title</i>	<i>Varchar</i>	255	Meta judul
5	<i>Meta_keyword</i>	<i>Text</i>		Meta kata kunci
6	<i>Meta_description</i>	<i>Text</i>	-	Meta desripsi

b. Struktur Tabel Rasa

Tabel ini digunakan untuk menyimpan data rasa. struktur tabel rasa dapat dilihat pada tabel 3.15 berikut ini.

Nama database : db\_sherli

Nama tabel : Rasa

Primary key : *rasa\_id*

Media penyimpanan : *Harddisk*



**Tabel 3.15** Struktur Tabel Rasa

<b>No</b>	<b>Nama Field</b>	<b>Type</b>	<b>Size</b>	<b>Keterangan</b>
1	rasa_id	<i>Int</i>	11	Id warna
2	rasa_name	<i>Varchar</i>	255	Nama warna

## c. Struktur Tabel Alamat

Tabel ini digunakan untuk menyimpan data alamat. Struktur tabel alamat dapat dilihat pada tabel 3.16 dibawah ini.

Nama database : db\_sherli  
 Nama tabel : alamat  
 Primary key : alamat\_id

Media penyimpanan : *Harddisk*

**Tabel 3.16** Alamat

<b>No</b>	<b>Nama Field</b>	<b>Type</b>	<b>Size</b>	<b>Keterangan</b>
1	alamat_id	<i>Int</i>	11	Id warna
2	alamat_name	<i>Varchar</i>	100	Nama warna

d. Struktur Tabel *Customer*

Tabel ini digunakan untuk menyimpan data Pelanggan. Struktur tabel customer dapat dilihat pada tabel 3.17 dibawah ini.

Nama database : db\_sherli  
 Nama tabel : *Customer*  
 Primary key : *Customer\_id*

Media penyimpanan : *Harddisk*

**Tabel 3.17** *Customer*

<b>No</b>	<b>Nama Field</b>	<b>Type</b>	<b>Size</b>	<b>Keterangan</b>
1	<i>Cust_id</i>	<i>Int</i>	11	Id pelanggan

2	<i>Cust_name</i>	<i>Varchar</i>	100	Nama pelanggan
3	<i>Cust_cname</i>	<i>Varchar</i>	100	Nama pendek pelanggan
4	<i>Cust_email</i>	<i>Varchar</i>	100	Email pelanggan
5	<i>Cust_phone</i>	<i>Varchar</i>	50	Nomor telpon pelanggan
6	<i>Cust_country</i>	<i>Int</i>	11	Negara pelanggan
7	<i>Cust_address</i>	<i>Text</i>	-	Alamat pelanggan
8	<i>Cust_city</i>	<i>Varchar</i>	100	Kota pelanggan
9	<i>Cust_state</i>	<i>Varchar</i>	100	Status pelanggan
10	<i>Cust_zip</i>	<i>Varchar</i>	30	Kode pos pelanggan
11	<i>Cust_b_name</i>	<i>Varchar</i>	100	Nama pelanggan
12	<i>Cust_b_cname</i>	<i>Varchar</i>	100	Nama pendek pelanggan
13				
14	<i>Cust_b_phone</i>	<i>Varchar</i>	100	No telpon pelanggan
15	<i>Cust_b_country</i>	<i>Int</i>	50	Negara pelnggan
16	<i>Cust_b_address</i>	<i>Text</i>	11	Alamat pelanggan
17	<i>Cust_b_city</i>	<i>Varchar</i>	-	Kota pelanggan
18	<i>Cust_b_state</i>	<i>Varchar</i>	100	Status pelanggan
19	<i>Cust_b_zip</i>	<i>Varchar</i>	100	Kode pelanggan
20	<i>Cust_s_name</i>	<i>Varchar</i>	30	Nama pelanggan
21	<i>Cust_s_cname</i>	<i>Varchar</i>	100	Nama pendek pelanggan

22	<i>Cust_s_phone</i>	<i>Varchar</i>	50	Nomor telepon pelanggan
23	<i>Cust_s_country</i>	<i>Int</i>	11	Negara pelanggan
24	<i>Cust_s_address</i>	<i>Text</i>	-	Alamat pelanggan
25	<i>Cust_s_city</i>	<i>Varchar</i>	100	Kota pelanggan
26	<i>Cust_s_state</i>	<i>Varchar</i>	30	Status pelanggan
27	<i>Cust_s_zip</i>	<i>Varchar</i>	100	Kode post pelanggan
28	<i>Cust_password</i>	<i>Varchar</i>	255	Password pelanggan
29	<i>Cust_datetime</i>	<i>Varchar</i>	100	Tanggal dan waktu
30	<i>Cust_timestamp</i>	<i>Varchar</i>	100	Cap waktu pelanggan
31	<i>Cust_status</i>	<i>Int</i>	1	Status pelanggan

e. Struktur Tabel *Customer Message*

Tabel ini digunakan untuk menyimpan data Pelanggan. Struktur tabel *customer message* dapat dilihat pada tabel 3.18 berikut ini.

Nama database : *db\_sherli*

Nama tabel : *Custemer message*

*Primary key* : *Custemer\_message\_id*

Media penyimpanan : *Harddisk*

**Tabel 3.18** *Custemer Message*

<b>No</b>	<b>Nama Field</b>	<b>Type</b>	<b>Size</b>	<b>Keterangan</b>
1	<i>Customer_message_id</i>	<i>Int</i>	11	Id pesanan pelanggan
2	<i>Subject</i>	<i>Varchar</i>	255	<i>subject</i>
3	<i>Message</i>	<i>Text</i>	-	pesan
4	<i>Order detail</i>	<i>Text</i>	-	Detail pemesanan
5	<i>Cust_id</i>	<i>Int</i>	11	Id pelanggan

f. Struktur Tabel *End Category*

Tabel ini digunakan untuk menyimpan data Kategori. Struktur tabel end category dapat dilihat pada tabel 3.19 dibawah ini.

Nama database : db\_sherli

Nama tabel : *end\_category*

Primary key : *ecat\_id*

Media penyimpanan : *Harddisk*

**Tabel 3.19** *End Category*

<b>No</b>	<b>Nama Field</b>	<b>Type</b>	<b>Size</b>	<b>Keterangan</b>
1	<i>Ecat_id</i>	<i>Int</i>	11	<i>Id end category</i>
2	<i>Ecat_name</i>	<i>Varchar</i>	255	Nama end category
3	<i>Mcat_id</i>	<i>Int</i>	11	<i>Category id pemesanan</i>

g. Struktur *table Language*

Tabel ini digunakan untuk menyimpan data Bahasa. Struktur tabel *language* dapat dilihat pada tabel 3.20 dibawah ini.

Nama database : db\_sherli

Nama tabel : *Language*

*Primary key* : Lang\_id

Media penyimpanan : *Harddisk*

**Tabel 3.20** *Language*

No	Nama Field	Type	Size	Keterangan
1	<i>Lang_id</i>	<i>Int</i>	11	Id bahasa
2	<i>Lang_name</i>	<i>Varchar</i>	255	Nama bahasa
3	<i>Lang_valuaes</i>	<i>Text</i>	-	Nilai bahasa

h. Struktur Tabel *Mid Category*

Tabel ini digunakan untuk menyimpan data category produk. Struktur tabel *mid category* dapat dilihat pada tabel 3.21 dibawah ini.

Nama database : db\_sherli

Nama tabel : *mid category*

*Primary key* : *Mcat\_id*

Media penyimpanan : *Harddisk*

**Tabel 3.21** *Mid Category*

No	Nama Field	Type	Size	Keterangan
1	<i>mcat_id</i>	<i>Int</i>	11	<i>Id category</i>
2	<i>mcat_name</i>	<i>Varchar</i>	255	Nama <i>category</i>
3	<i>Tcat_id</i>	<i>Int</i>	11	Top category tabel id

i. Struktur Tabel *Order*

Tabel ini digunakan untuk menyimpan data order. Struktur tabel order dapat dilihat pada tabel 3.22 dibawah ini.

Nama database : db\_sherli

Nama tabel : *order*

*Primary key* : *id*

Media penyimpanan : *Harddisk*

**Table 3.22** Tabel *Order*

No	Nama Field	Type	Size	Keterangan
1	Id	<i>Int</i>	11	Id pelanggan
2	Produk_id	<i>Int</i>	11	Id produk
3	<i>Produk_name</i>	<i>Varchar</i>	255	Nama produk
4	<i>size</i>	<i>Varchar</i>	100	Ukuran
5	<i>colour</i>	<i>Varchar</i>	100	Warna
6	<i>quantity</i>	<i>Varchar</i>	50	Jumlah
7	<i>Unit_price</i>	<i>Varchar</i>	50	Unit harga
8	<i>Paymen_id</i>	<i>Varchar</i>	255	Id pembayaran

j. Struktur Tabel *Payment*

Tabel ini digunakan untuk menyimpan data payment. Struktur tabel payment dapat dilihat pada tabel 3.23 berikut ini.

Nama database : db\_sherli

Nama tabel : *Payment*

*Primary key* : *id*

Media penyimpanan : *Harddisk*

**Tabel 3.23 Payment**

<b>No</b>	<b>Nama Field</b>	<b>Type</b>	<b>Size</b>	<b>Keterangan</b>
1	<i>id</i>	<i>Int</i>	11	Id pembayaran
2	<i>Customer_id</i>	<i>Int</i>	11	Id pelanggan
3	<i>Customer_name</i>	<i>Varchar</i>	255	Nama pelanggan
4	<i>Custemer_email</i>	<i>Varchar</i>	255	Email pelanggan
5	<i>Payment_date</i>	<i>Varchar</i>	50	Waktu pembayaran
6	<i>Txnid</i>	<i>Varchar</i>	255	Id transaksi
7	<i>Paid_amount</i>	<i>Int</i>	11	Jumlah pembayaran
8	<i>Card_number</i>	<i>Varchar</i>	50	Nomor kartu
9	<i>Card_cvv</i>	<i>Varchar</i>	10	Kartu cvv
10	<i>Card_month</i>	<i>Varchar</i>	10	Card month
11	<i>Card_year</i>	<i>Varchar</i>	10	Card year
12	<i>Bank_transacion_info</i>	<i>Text</i>	-	Info bank transaksi
13	<i>Payment_metode</i>	<i>Varchar</i>	20	Metode pembayaran
14	<i>Payment_status</i>	<i>Varchar</i>	25	Status pembayaran
15	<i>Shipping_status</i>	<i>Varchar</i>	20	Status pengiriman
16	<i>Payment_id</i>	<i>Varchar</i>	255	Id pembayaran

## k. Struktur Tabel Produk

Tabel ini digunakan untuk menyimpan data produk. Struktur tabel produk dapat dilihat pada tabel 3.24 dibawah ini.

Nama database : db\_sherli  
 Nama tabel : Tbl\_produk  
 Primary key : P\_id  
 Media penyimpanan : Harddisk

**Tabel 3.24** Produk

No	Nama Field	Type	Size	Keterangan
1	<i>P_id</i>	<i>Int</i>	11	Id produk
2	Kode_perusahaan	<i>Varchar</i>	11	Kode perusahaan
3	<i>P_name</i>	<i>Varchar</i>	255	Nama produk
4	<i>P_old_price</i>	<i>Varchar</i>	10	Harga lama
5	<i>P_curent_price</i>	<i>Varchar</i>	10	Harga sekarang
6	<i>P_qty</i>	<i>Int</i>	10	jumlah
7	<i>P_Featur_photo</i>	<i>Varchar</i>	255	Fitur poto
8	<i>P_description</i>	<i>Text</i>	-	deskripsi
9	<i>P_short_description</i>	<i>Text</i>	-	Deskripsi pendek
10	<i>P_Featur</i>	<i>Text</i>	-	fitur
11	<i>P_condition</i>	<i>Text</i>	-	Kondisi
12	<i>P_return_policy</i>	<i>Text</i>	-	Kebijakan pengembalian
13	<i>P_total_fiew</i>	<i>Int</i>	11	Total pengunjung
14	<i>P_is_featured</i>	<i>Int</i>	1	ditampilkan



15	<i>P_is_active</i>	<i>Int</i>	1	aktif
16	<i>Ecat_id</i>	<i>Int</i>	11	Id kategori

### 1. Struktur Table Produk Rasa

Tabel ini digunakan untuk menyimpan data rasa produk. Struktur tabel produk rasa dapat dilihat pada tabel 3.25 dibawah ini.

Nama database : db\_sherli  
 Nama tabel : *Tbl\_produk\_colour*  
 Primary key : id  
 Media penyimpanan : *Harddisk*

**Tabel 3.25** produk rasa

No	Nama Field	Type	Size	Keterangan
1	<i>id</i>	<i>Int</i>	11	Id
2	<i>Colour_id</i>	<i>Int</i>	11	Id warna
3	<i>P_id</i>	<i>Int</i>	11	Id produk

### m. Struktur tabel produk photo

Tabel ini digunakan untuk menyimpan data foto produk. Struktur tabel produk foto dapat dilihat pada gambar 3.26 berikut ini.

Nama database : db\_sherli  
 Nama tabel : *Tbl\_produk\_Photo*  
 Primary key : *Pp\_id*  
 Media penyimpanan : *Harddisk*

**Tabel 3.26** Produk Photo

No	Nama Field	Type	Size	Keterangan
1	<i>Pp_id</i>	<i>Int</i>	11	Id photo
2	<i>photo</i>	<i>Varchar</i>	255	foto

3	<i>P_id</i>	<i>Int</i>	11	Id produk
---	-------------	------------	----	-----------

n. Struktur Tabel Produk *Size*

Tabel ini digunakan untuk menyimpan data ukuran produk. Struktur tabel produk size dapat dilihat pada tabel 3.27 dibawah ini.

Nama *database* : db\_sherli

Nama tabel : Tbl\_produk\_size

*Primary key* : -

Media penyimpanan : *Harddisk*

**Tabel 3.27** Produk *Size*

No	Nama Field	Type	Size	Keterangan
1	Id	<i>Int</i>	11	Id
2	<i>Size_id</i>	<i>Int</i>	11	Id ukuran
3	P_id	<i>Int</i>	11	Id produk

o. Struktur Tabel Tbl Produk *Supplier*

Tabel ini digunakan untuk menyimpan data *supplier*. Struktur tabel produk *supplier* dapat dilihat pada tabel 3.28 berikut ini.

Nama database : db\_sherli

Nama tabel : Tbl\_produk\_supplier

*Primary key* : P\_id

Media penyimpanan : *Harddisk*

**Tabel 3.28** *Supplier*

No	Nama Field	Type	Size	Keterangan
1	P_id	<i>Int</i>	11	Id produk
2	P_kode	<i>Varchar</i>	30	Kode produk

3	P_perusahaan	Varchar	30	Kode perusahaan
4	P_name	Varchar	30	Nama produk
5	P_current_price	Varchar	30	Harga baru
6	P_qty	Varchar	3	Jumlah produk
7	P_photo	Varchar	60	Poto produk
8	P_description	Text	-	Deskripsi produk
9	P_sup_id	Varchard	12	Id produk supplier

p. Struktur Tabel Tbl\_Rating

Tabel ini digunakan untuk menyimpan data rating. Struktur tabel rating dapat dilihat pada tabel 3.29 dibawah ini.

Nama database : db\_sherli

Nama tabel : Tbl\_rating

Primary key : -

Media penyimpanan : Harddisk

**Tabel 3.29** rating

No	Nama Field	Type	Size	Keterangan
1	Rt_id	Int	11	Id rating
2	P_id	Int	11	Id produk
3	Cust_id	Int	11	Id pelanggan
4	Coment	Text	-	Comment
5	Rating	Int	11	rating

q. Struktur Tabel *Setting*

Tabel ini digunakan untuk menyimpan data seting Pada admin. Struktur tabel seting dapat dilihat pada tabel 3.30 dibawah ini.

Nama database : db\_sherli

Nama tabel : *Tbl\_setting*

*Primary key* : -

Media penyimpanan : *Harddisk*

**Tabel 3.30** *Seting*

<b>No</b>	<b>Nama Field</b>	<b>Type</b>	<b>Size</b>	<b>Keterangan</b>
1	<i>Id</i>	<i>Int</i>	11	Id seting
2	<i>Logo</i>	<i>varchar</i>	255	logo
3	<i>Favicon</i>	<i>Varchar</i>	255	favicon
4	<i>Foother_about</i>	<i>Text</i>	-	Tentang dibawah
5	<i>Foother_copyright</i>	<i>Text</i>	-	copyright
6	<i>Contact_addres</i>	<i>Text</i>	-	Alamat kontak
7	<i>Contact_email</i>	<i>Varchar</i>	255	Alamat email
8	<i>Contact_phone</i>	<i>Varchar</i>	255	Alamat telepon
9	<i>Contact_fax</i>	<i>Varchar</i>	255	Alamat fax
10	<i>Contact_map_iframe</i>	<i>Text</i>	-	Alamat google
11	<i>Receive_email</i>	<i>Varchar</i>	255	Pegiriman email
12	<i>Receive_email_subject</i>	<i>Varchar</i>	255	Subjek pengiriman email
13	<i>Receive_email_thank_you_me</i>	<i>Text</i>	-	Pesan email

	<i>ssage</i>			terimakasih
14	<i>Forget_passwort_message</i>	<i>Text</i>	-	Lupa paswort
15	<i>Total_recent_post_footer</i>	<i>Int</i>	10	Total fother
16	<i>Total_popular_post_footer</i>	<i>int</i>	10	Total popular post footer
17	<i>Total_recent_post_sidebar</i>	<i>int</i>	11	Total post sidebar
18	<i>Total_popular_post_sidebar</i>	<i>Int</i>	11	Total popular post sidebar
19	<i>Total_featured_produk_home</i>	<i>Int</i>	11	Total featured produk home
20	<i>Total_lates_produk_home</i>	<i>Int</i>	11	Total lates produk home
21	<i>Total_popular_produk_home</i>	<i>Int</i>	11	Total popular produk home
22	<i>Meta_title_home</i>	<i>text</i>	-	Meta title home
23	<i>Meta_keyword_home</i>	<i>text</i>	-	Meta keyword home
24	<i>Meta_description_home</i>	<i>Text</i>	-	Meta description home
25	<i>Banner_login</i>	<i>Varchar</i>	255	Banner login
26	<i>Banner_registration</i>	<i>Varchar</i>	255	<i>Banner registration</i>
27	<i>Banner_forget_fasword</i>	<i>varchar</i>	255	<i>Banner forget fasword</i>
28	<i>Banner_reset_faswod</i>	<i>varchar</i>	255	<i>Banner reset</i>

				<i>faswod</i>
29	<i>Banner_search</i>	<i>varchar</i>	255	<i>Banner search</i>
30	<i>Banner_cart</i>	<i>Varchar</i>	255	<i>Banner cart</i>
31	<i>Banner_chekout</i>	<i>Varchar</i>	255	<i>Banner chekout</i>
32	<i>Banner_product_category</i>	<i>Varchar</i>	255	<i>Banner product category</i>
33	<i>Banner_blog</i>	<i>Varchar</i>	255	<i>Banner blog</i>
34	<i>Cta_title</i>	<i>Varchar</i>	255	<i>Cta title</i>
35	<i>Cta_conten</i>	<i>text</i>	-	<i>Cta conten</i>
36	<i>Cta_read_more_text</i>	<i>Varchar</i>	255	<i>Cta read more text</i>
37	<i>Cta_read_more_url</i>	<i>Text</i>	-	<i>Cta read more url</i>
38	<i>Cta_photo</i>	<i>Varchar</i>	255	<i>Cta photo</i>
39	<i>Featured_produk_title</i>	<i>varchar</i>	255	<i>Featured produk title</i>
40	<i>Featured_produk_subtitle</i>	<i>Varchar</i>	255	<i>Featured produk subtitle</i>
41	<i>Last_produk_title</i>	<i>Varchar</i>	255	<i>Last produk title</i>
42	<i>Last_produk_subtitle</i>	<i>Varchar</i>	255	<i>Last produk subtitle</i>
43	<i>Popular_produk_title</i>	<i>Varchar</i>	255	<i>Popular produk title</i>
44	<i>Popular_produk_subtitle</i>	<i>Varchar</i>	255	<i>Popular</i>

				<i>produk subtitle</i>
45	<i>Testimonial_title</i>	<i>Varchar</i>	255	<i>Testimonial title</i>
46	<i>Testimonial_subtitle</i>	<i>Varchar</i>	255	<i>Testimonial subtitle</i>
47	<i>Testimonial_photo</i>	<i>Varchar</i>	255	<i>Testimonial photo</i>
48	<i>Blog_title</i>	<i>Varchar</i>	255	<i>Blog title</i>
49	<i>Blog_subtitle</i>	<i>Varchar</i>	255	<i>Blog subtitle</i>
50	<i>Newsletter_text</i>	<i>Text</i>	-	<i>Newsletter text</i>
51	<i>Paypal_email</i>	<i>Varchar</i>	255	<i>Paypal email</i>
52	<i>Stripe_public_key</i>	<i>Varchar</i>	255	<i>Stripe public key</i>
53	<i>Stripe_secret_key</i>	<i>Varchar</i>	255	<i>Stripe secret key</i>
54	<i>Bank_detail</i>	<i>text</i>	-	<i>Bank detail</i>
55	<i>Before_head</i>	<i>text</i>	-	<i>Before head</i>
56	<i>After_body</i>	<i>text</i>	-	<i>After body</i>
57	<i>Before_body</i>	<i>text</i>	-	<i>Before body</i>
58	<i>Home_service_on_off</i>	<i>Int</i>	11	<i>Home service on off</i>
59	<i>Home_welcome_on_off</i>	<i>int</i>	11	<i>Home welcome on off</i>
60	<i>Home_Featur_product_on_off</i>	<i>Int</i>	11	<i>Home Featur product on off</i>
61	<i>Home_Lates_product_on_off</i>	<i>Int</i>	11	<i>Home Lates</i>

				<i>product on off</i>
62	<i>Home_Popular_product_on_off</i>	<i>Int</i>	11	<i>Home Popular product on off</i>
63	<i>Home_testimonial_on_off</i>	<i>Int</i>	11	<i>Home testimonial on off</i>
64	<i>Home_blog_on_off</i>	<i>Int</i>	11	<i>Home blog on off</i>
65	<i>Newsletter_on_off</i>	<i>Int</i>	11	<i>Newsletter on off</i>

r. Struktur Tabel *Tbl\_Size*

Tabel ini digunakan untuk menyimpan data Ukuran. Struktur tabel *size* dapat dilihat pada tabel 3.31 dibawah ini.

Nama database : *db\_sherli*

Nama tabel : *Tbl\_Size*

*Primary key* : -

Media penyimpanan : *Harddisk*

**Tabel 3.31** *size*

<b>No</b>	<b>Nama Field</b>	<b>Type</b>	<b>Size</b>	<b>Keterangan</b>
1	<i>Size_id</i>	<i>Int</i>	11	Id ukuran
2	<i>Size_name</i>	<i>Varchar</i>	255	Nama ukuran

s. Struktur Tabel *Tbl\_Supplier*

Tabel ini digunakan untuk menyimpan data supplier. Struktur tabel supplier dapat dilihat pada tabel 3.32 berikut ini.

Nama database : *db\_sherli*

Nama tabel : *Tbl\_supplier*



Primary key : -

Media penyimpanan : Harddisk

**Tabel 3.32 Supplier**

No	Nama Field	Type	Size	Keterangan
1	id	Int	11	Id supplier
2	No_pt	Varchar	10	No pt
3	Full_name	Varchar	30	Nama
4	Perusahaan	Varchar	255	Perusahaan
5	email	Varchar	255	Email
6	phone	Varchar	255	Telepon
7	paswort	Varchar	255	paswort
8	Photo	Varchar	255	Photo
9	Role	Varchar	30	Role
10	status	Varchar	11	status

t. Struktur Tabel *Tbl\_Top\_Categori*

Tabel ini digunakan untuk menyimpan data top kategori produk. Struktur tabel top kategori dapat dilihat pada tabel 3.33 dibawah ini.

Nama database : db\_sherli

Nama tabel : *Tbl\_Top\_categori*

Primary key : Tcat\_id

Media penyimpanan : Harddisk

**Tabel 3.33 Tbl\_Top\_Categori**

No	Nama Field	Type	Size	Keterangan
1	Tcat_id	Int	11	Id
2	Tcat_name	Varchar	255	Nama
3	Show_on_menu	Int	1	Tampilkan di menu

u. Struktur tabel *tbl\_user*

Tabel ini digunakan untuk menyimpan data *User*. Struktur tabel *user* dapat dilihat pada tabel 3.34 dibawah ini.

Nama database : db\_sherli

Nama tabel : *Tbl\_User*

Primary key : id

Media penyimpanan : *Harddisk*

**Tabel 3.34** *User*

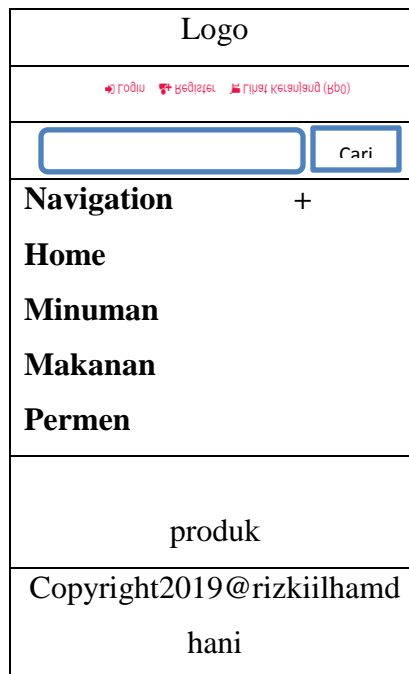
No	Nama Field	Type	Size	Keterangan
1	id	<i>Int</i>	11	Id
2	<i>Full_name</i>	<i>Varchar</i>	100	Nama
3	Perusahaan	<i>Varchar</i>	30	perusahaan
4	<i>email</i>	<i>Varchar</i>	255	email
5	<i>phone</i>	<i>Varchar</i>	100	telepon
6	<i>password</i>	<i>Varchar</i>	255	password
7	<i>Photo</i>	<i>Varchar</i>	255	Poto
8	<i>Role</i>	<i>Varchar</i>	30	Peran
9	status	<i>Varchar</i>	10	status

### 3.1.3.19 Rancangan Interface Pada Sistem Aplikasi

#### 3.1.3.19.1 Rancangan Interface Pada Halaman Halaman Pelanggan

Tampilan halaman pelanggan ketika mengakses sistem serta penjelasan menu-menu pada halaman *user*. pada gambar menu login berfungsi untuk masuk ketika *user* sudah membuat akun di *register*. Pada menu keranjang berfungsi sebagai penampung barang yang dipilih oleh *user*. Pada menu *navigasi* berfungsi sebagai penunjuk halaman yang ada diantaranya home, minuman, makan, permen. Menu *home* berfungsi sebagai tampilan awal ketika pengunjung mengakses halaman utama. Menu minuman berfungsi menampilkan jenis minuman, menu makanan berfungsi

menampilkan makanan yang dipilih. Menu permen berfungsi untuk menampilkan produk permen. Pada posisi atas terdapat logo berfungsi untuk menambahkan logo di tampilan pengunjung. Halaman pelanggan merupakan halaman yang akan ditampilkan jika pelanggan mengaksesnya dapat dilihat pada gambar 3.12 dibawah ini.















**Gambar 3.12** Tampilan pengunjung

Pada gambar menu *login* berfungsi untuk masuk ketika *user* sudah membuat akun di *register*. Pada menu keranjang berfungsi sebagai penampung barang yang dipilih oleh *user*. Pada menu navigasi berfungsi sebagai penunjuk halaman yang ada diantaranya *home*, minuman, makan, permen. Menu *home* berfungsi sebagai tampilan awal ketika pengunjung mengakses halaman utama. Menu minuman berfungsi menampilkan jenis minuman, menu makanan berfungsi menampilkan makanan. Menu permen berfungsi untuk menampilkan produk permen. Pada posisi atas terdapat logo berfungsi untuk menambahkan logo di tampilan pengunjung.

### 3.1.3.19.2 Rancangan *Interface* Pada Halaman Admin

Halaman Admin merupakan halaman yang akan ditampilkan jika admin mengakses halaman admin dan berhasil masuk ke halaman admin. Setelah admin berhasil masuk ke sistem admin dapat melakukan *insert,edit,delete*. Rancangan tampilan halaman admin dapat dilihat pada gambar 3.13 dibawah ini.

<b>SherlySnack</b> <b>Admin Panel</b> 		
 <b>Dasbord</b>	<b>Tampilan menu yang dipilih</b>	
 <b>Setting</b>		
 <b>Shop section</b>		
 <b>Produk</b>		
 <b>Supplier</b>		
 <b>Order</b>		
 <b>Rating</b>		
 <b>Bahasa Seting</b>		
 <b>Pesan</b>		
 <b>Pelanggan</b>		

**Gambar 3.13** Tampilan Admin

Pada rancangan tampilan utama admin terdapat beberapa pilihan menu yaitu menu *dashboard* berfungsi untuk menampilkan halaman utama admin ketika berhasil login, menu *setting* berfungsi untuk *setting* di halaman pengunjung, diantaranya *setting logo, vapicon, banner, email, home page*. Menu *shop section* berfungsi untuk mengatur ukuran produk, rasa, daerah dan kategori produk. Menu produk berfungsi untuk tambah produk, ubah produk, edit produk. Menu supplier berfungsi sebagai tambah *supplier*. Menu *order* berfungsi untuk melihat pesanan pelanggan. Menu rating berfungsi untuk melihat rating produk yang diberikan oleh pelanggan. Menu pesan berfungsi untuk melihat riwayat pesan ke pelanggan. Menu pelanggan berfungsi

untuk melihat pelanggan yang aktif dan dapat mengubah status pelanggan apakah aktif atau tidak. Di sisi kanan atas akan tampil foto dan nama admin yang login ke sistem, admin dapat mengedit profil diantaranya *update* informasi, *update* password, *update* photo, *update* telepon, *update* email address.

### 3.1.3.19.3 Rancangan Interface Pada Halaman *Supplier*

Halaman *supplier* merupakan halaman yang akan ditampilkan jika *supplier* mengakses halaman *supplier* dan berhasil masuk ke halaman *supplier*. Rancangan *interface* halaman *supplier* dapat dilihat pada gambar 3.14 dibawah ini.

		(foto)
<b>Supplier Panel</b>		
→ <b>Dasboard</b>	<b>Selamat datang supplier</b>	
→ <b>Produk</b>		

**Gambar 3.14** Tampilan *Supplier*

Pada rancangan tampilan halaman *supplier* di sebelah kanan terdapat foto *supplier* dan nama *supplier* ketika *supplier* berhasil login ke sistem dan di tampilan seperti gambar 3.14 diatas.

### 3.1.4 Pembentukan Prototipe

pada tahap ini peneliti melakukan pengkodean program yang sesuai dengan tahapan pemodelan perancangan secara cepat. Penulis menggunakan *tools sublime text 3* untuk pengkodean program. Bahasa pemrograman php sebagai pengolahan data ke database *phpmyadmin* serta memanfaatkan *tools framework bootstraps* untuk membuat tampilan yang menarik dan responsip. Lalu penulis menguji apakah fungsi menu yang ada di sistem berjalan sesuai dengan perancangan yang dibuat. Hasil yang dirancang oleh penulis dapat dilihat di bab iv hasil penelitian dan pembahasan.

### **3.1.5 Penyerahan Sistem Perangkat Lunak Ke Pelanggan/Pengguna**

setelah pengkodean program dan pengujian selesai dilakukan tahapan selanjutnya yaitu penyerahan sistem yang dibuat kepada pengguna. Pada tahapan ini penulis menyerahkan sistem kepada pengguna untuk memanfaatkan sistem yang dibuat.

### **3.2 Proses Kerja Aplikasi**

Proses kerja aplikasi *e-service* terintegrasi yaitu dimulai dari admin menambahkan *supplier* lalu *supplier* menginputkan produk yang akan ditawarkan di pelanggan, *supplier* dapat melihat stok produk yang tersedia untuk memudahkan *supplier* memasok produk ketika produk habis. Admin dapat menambahkan produk untuk ditawarkan di pelanggan, melihat pesanan pelanggan, melihat rating produk. Pelanggan memesan produk dan mencari barang yang diinginkan di pencarian dengan menggunakan algoritma *brute force* yang berbasis *web mobile*.