KEBUTUHAN SISTEM

* 1. Fungsional
* Tools untuk Analisa Perancangan Sistem
* *UML (Unified Modelling Language)*

adalah sebuah bahasa yang berdasarkan grafik/gambar untuk mem*visual*isasi, menspesifikasikan, membangun, dan pendokumentasian dari sebuah sistem pengembangan *software* OO (*Object-Oriented*). UML sendiri juga memberikan standar penulisan sebuah sistem *blue print*, yang meliputi konsep bisnis proses, penulisan kelas-kelas dalam bahasa program yang spesifik, skema database, dan komponen-komponen yang diperlukan dalam sistem.

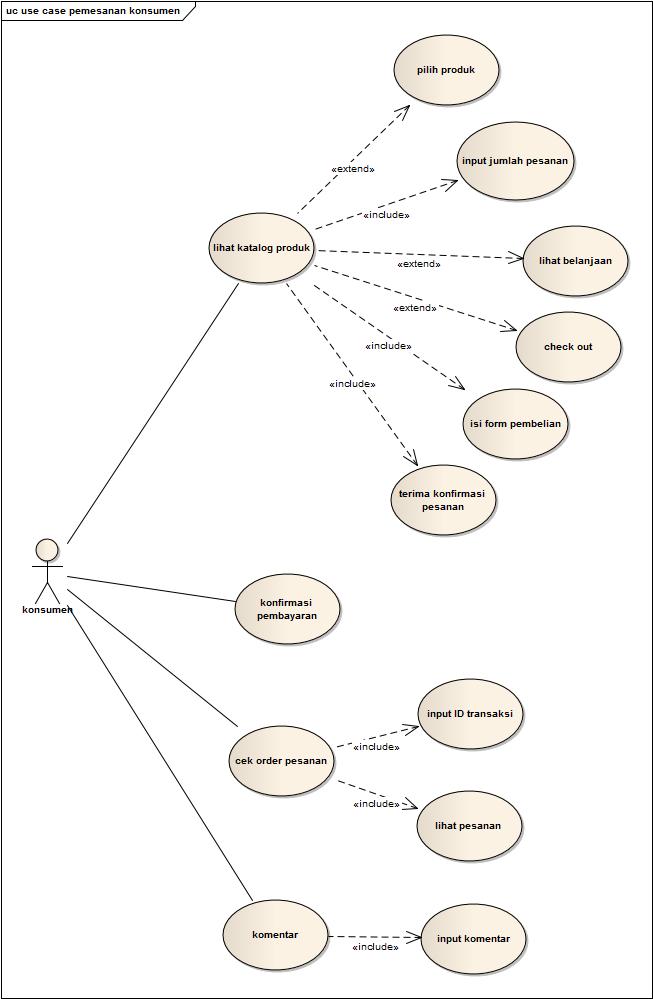
* + - 1. *Use Case Diagram*

*Use Case Diagram* merupakan diagram yang menggambarkan semua kasus (*case*) yang akan ditangani oleh perangkat lunakbeserta aktor dan pelakunya. Adapun aktor-aktor yang ada dalam *Use Case Diagram* yaitu:

1. Konsumen
2. *Owner*
3. Admin

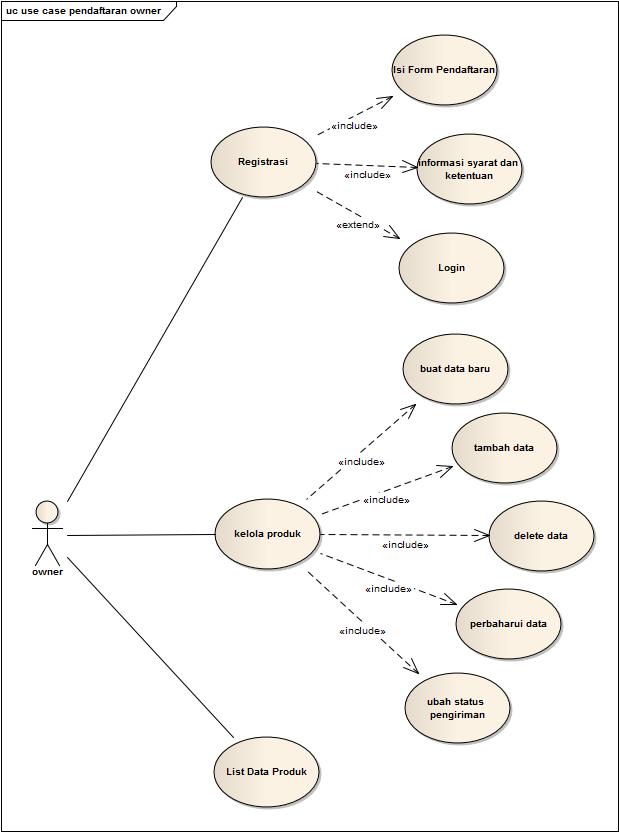
Dalam penerapan konsumen dalam pemesanan barang yang akan di beli dalan web ecommerce dengan menggunakan penjelasan *Use Case* dari msaing-masing aktor tersebut sebagai berikut :

1. *Use Case* PemesananKonsumen

****

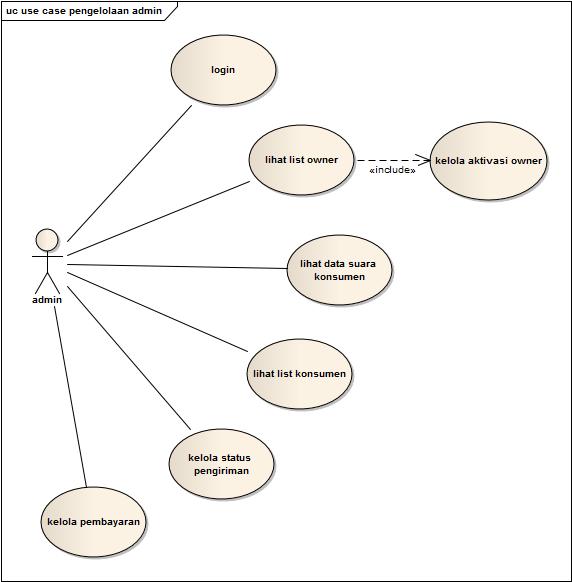
**Gambar 1. *Use Case* PemesananKonsumen**

1. *Use Case* Pendaftaran *Owner*



**Gambar 2. *Use Case* Pendaftaran *Owner***

1. *Use Case* Pengelolaan Admin



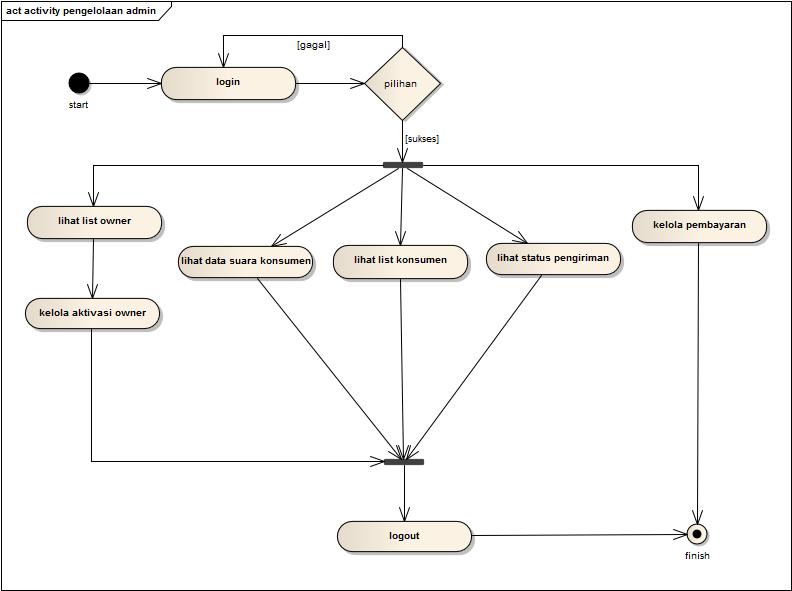
**Gambar 3. *Use Case* Pengelolaan Admin**

* + - 1. *Activity Diagram*

*Activity Diagram* ini berfungsi untuk mengilustrasikan proses berjalannya alur jual-beli *online*.

Adapun *activity* nya sebagai berikut:

1. *Activity* Pengelolaan Admin

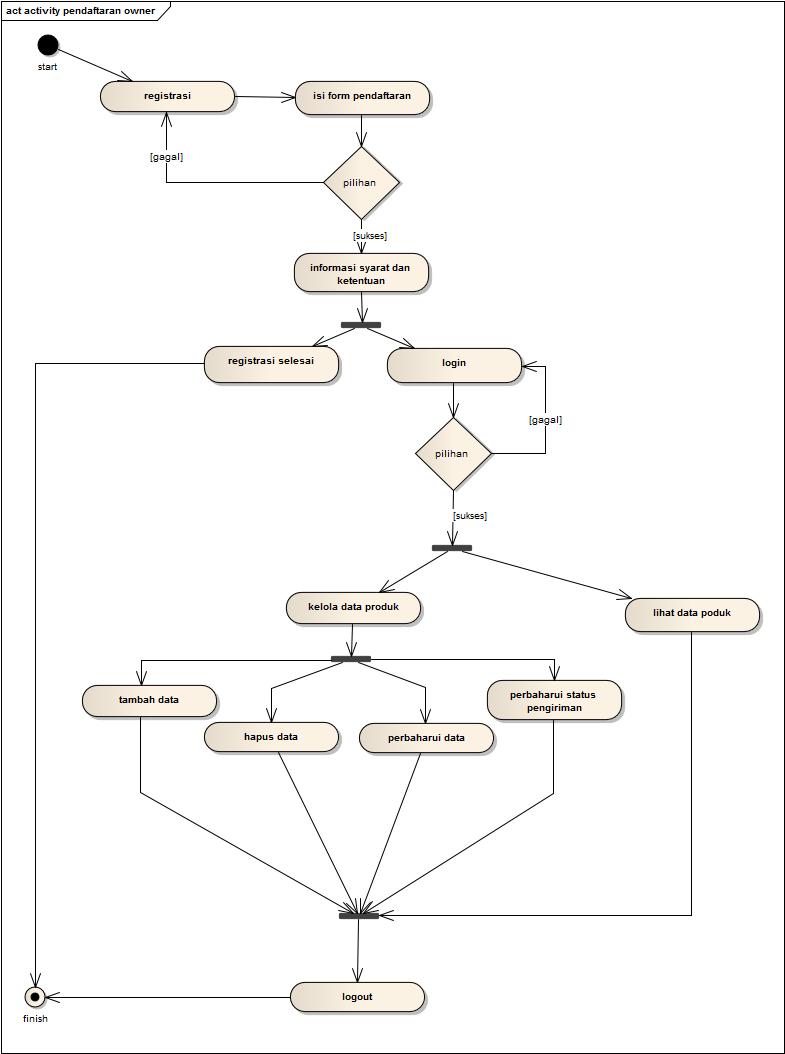
**

**Gambar 4. ActivityPengelolaan Admin**

**Keterangan :**

Admin memasukan usernamedan passworduntuk dapat mengelolawebsite. mengelolawebsite diantaranya adalah pengelolaan halaman pada website , memantau data-data dari setiap Owner dan konsumen, menanggapi komentar dari konsumen, dan mengelola keuangan.

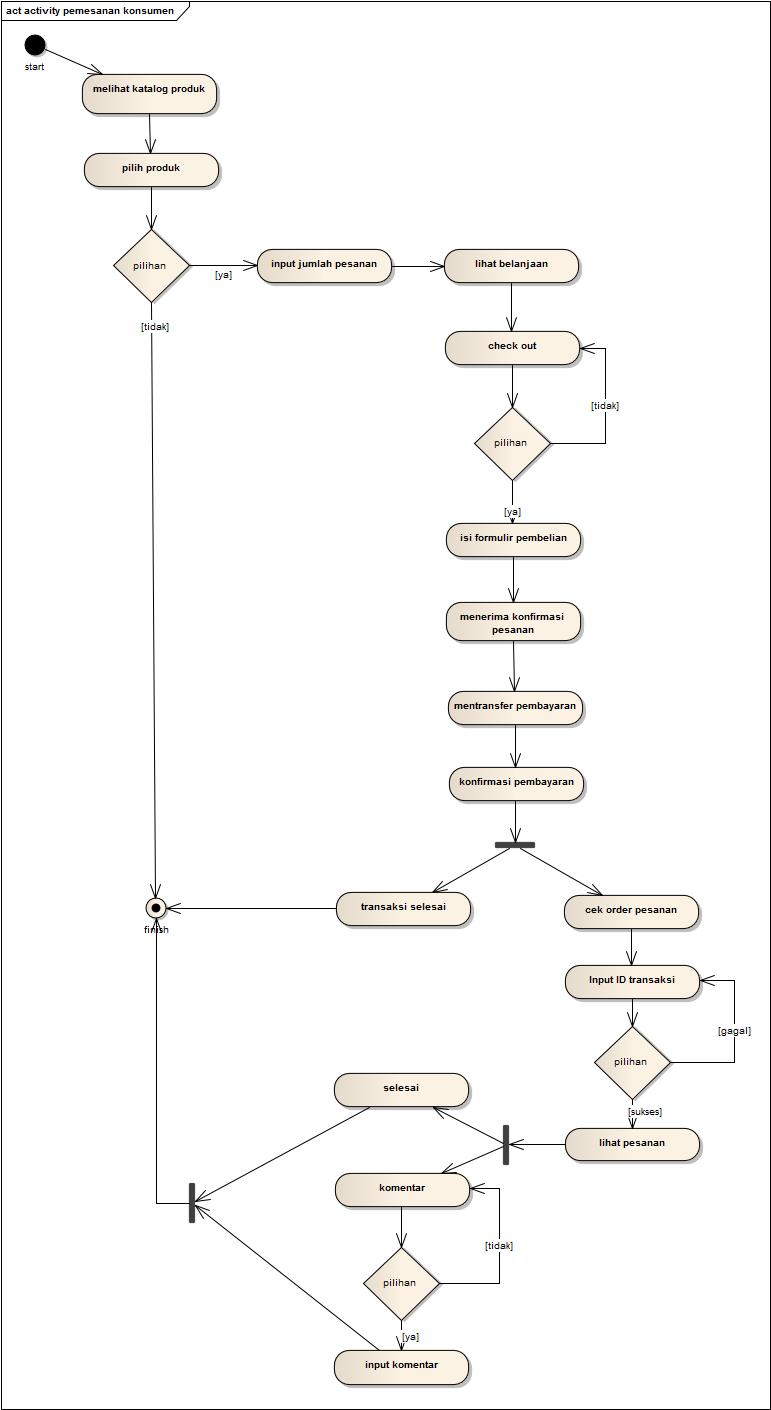
1. *Activity* Pendaftaran*Owner*

****

**Gambar 5. *Activity* Pendaftaran *Owner***

**Keterangan :**

*Owner* yang ingin mebuka aplikasi harus mendaftarkan diri terlebih dahulu dengan cara meng*input*kan data yang lengkap serta menentukan kategori toko yang akan dipilih. Setelah data lengkap maka secara otomatis *Owner* mendapatkan *username*dan *password*dan informasi syarat dan ketentuan lewat *email* konfirmasi. Lalu *username*dan *password* tersebut di*input*kan pada menu *Login*, setelah sukses lagin *Owner* tersebut dapat melakukan pengelolaan tokonya dari mengelola produk, merubah status *order*, hingga memanajemen pelanggan-pelanggan yang telah membeli produk di tokonya.

1. *Activity* Pemesanan Konsumen

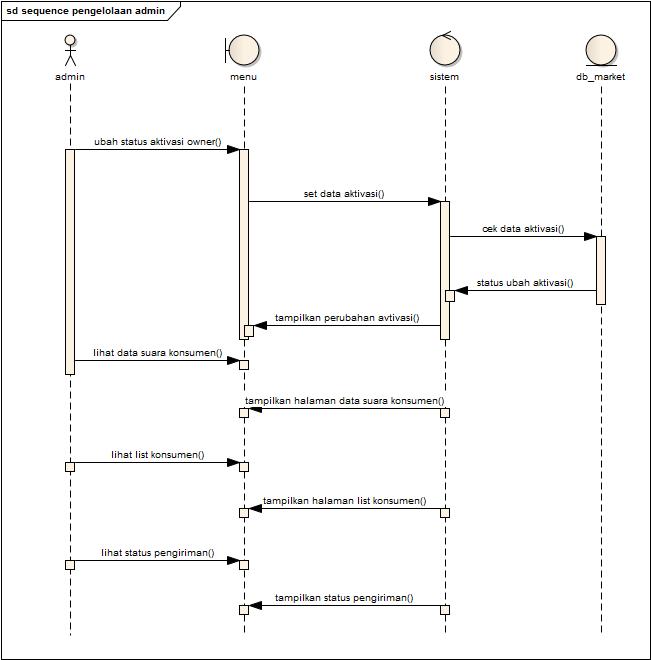
**Gambar 6. *Activity* Pemesanan Konsumen**

**Keterangan :**

Konsumen dapat melihat produk dan detail produk, lalu konsumen dapat membeli produk tersebut dengan mengisikan jumlah produk yang dipesan lalu mengklik tombol *add to chart*. Konsumen dapat melihat *list* belanjaannya. Jika selesai berbelanja konsumen dapat mengakhirinya dengan cara menklik tombol *Check Out* setelah itu konsumen mengisi *form* pembelian. Setelah selesai secara otomatis sistem akan mengirimkan *list* pemesanan ke halaman *Owner* dan sistem akan mengirimkan konfirmasi berupa *email* untuk konsumen yang didalamnya terdapat id transaksi. Id transaksi tersebut digunakan oleh konsumen untuk melihat *list* pemesanan dalam menu check *order* dan pada menu tersebut konsumen dapat memberi komentar. Setelah itu konsumen membayar produk yang dipesannya dengan cara mentransfer ke rekening *bank* yang tertera pada *email*. Lalu admin menerima transfer pembayaran tersebut dan mengkonfirmasikannya pada *Owner*, lalu *Owner* mengirimkan pesanan tersebut kepada konsumen. Jika pesanan tersebut telah diterima, maka *Owner* mengkonfirmasikannya pada admin.Admin menerima konfirmasi tersebut dan mengeCek ke konsumen. Jika sudah diterima oleh konsumen, maka admin akan mentransfer biaya pemesanan pada *Owner* dan setelah *Owner* menerima tranferan tersebut, *Owner* mengkonfirmasikan pada admin.

* + - 1. *Sequence Diagram*

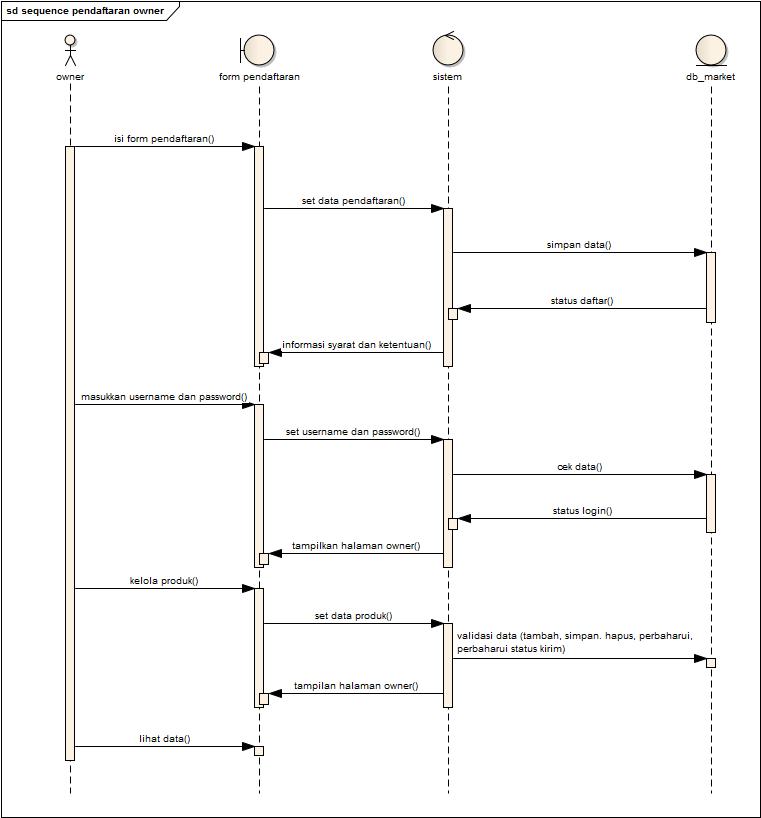
*Sequence Diagram* ini berfungsi untuk mengilustrasikan alur cerita secara sistematis.

1. *Sequence* Pengelolaan Admin

**Gambar 7. *Sequence*Pengelolaan Admin**

**Keterangan :**

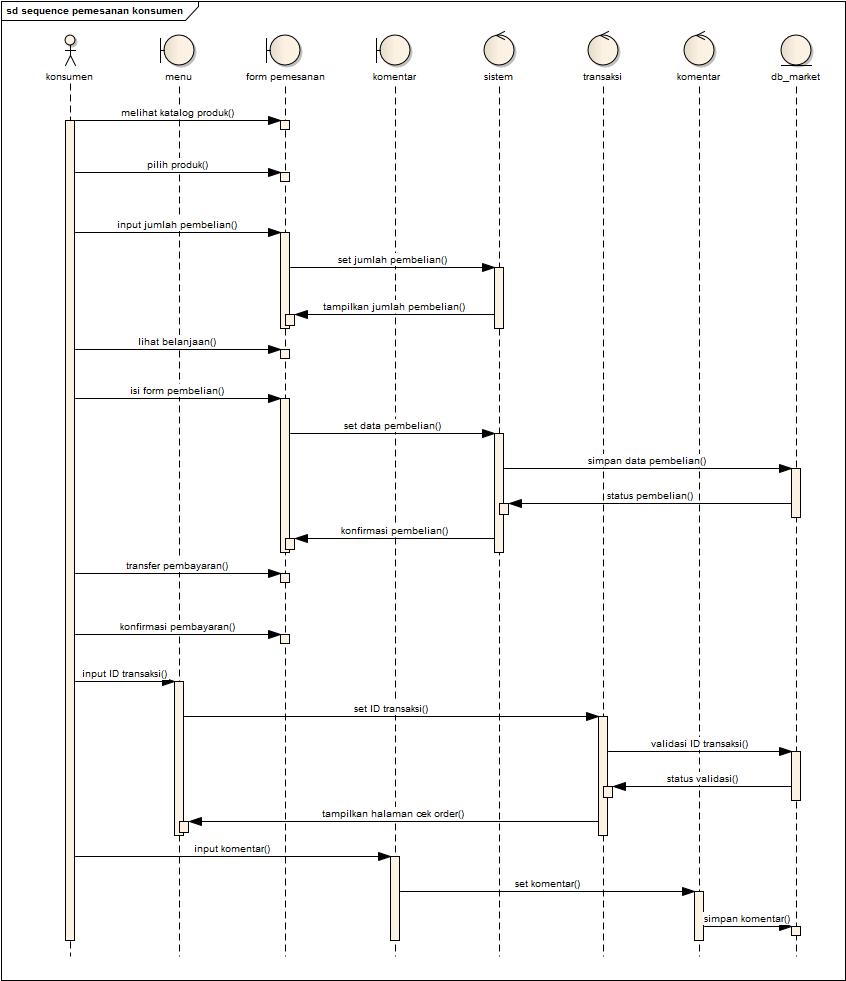
Admin mengubah status aktivasi owner lalu data di set oleh sistem dan di cek oleh *database*, *database*memberikan status pengecekan dan sistem menampilkan perubahan tersebut. Selain itu admin dapat melihat data suara konsumen, melihat list konsumen, melihat status pengiriman barang untuk mengetahui konsumen mana saja yang sudah menerima barang.

1. *Sequence* Pendaftaran *Owner*

**Gambar 8. *Sequence* Pendaftaran *Owner***

**Keterangan :**

Owner mengisi *form*pendaftaran lalu di set oleh sistem dan disimpan ke *database*, setelah itu *database* memberikan status penyimpanan data, lalu sistem mengirim email konfirmasi yang didalamnya terdapat username dan password dan info syarat dan ketentuan, lalu menampilkan menu login. Pada menu login owner meng*input*kan *username* dan *password*, lalu data yang di*input*kan akan diambil oleh sistem dan mem*valid*asi data yang berada pada database. Jika data *valid*, maka sistem akan menampilkan halaman *website*. Ownerpun dapat mengelola produk di tiap tokonya masing – masing, dari menambah produk, mengubah data produk, sampai mengubah status pengiriman barang.

1. *Sequence* Pemesanan Konsumen

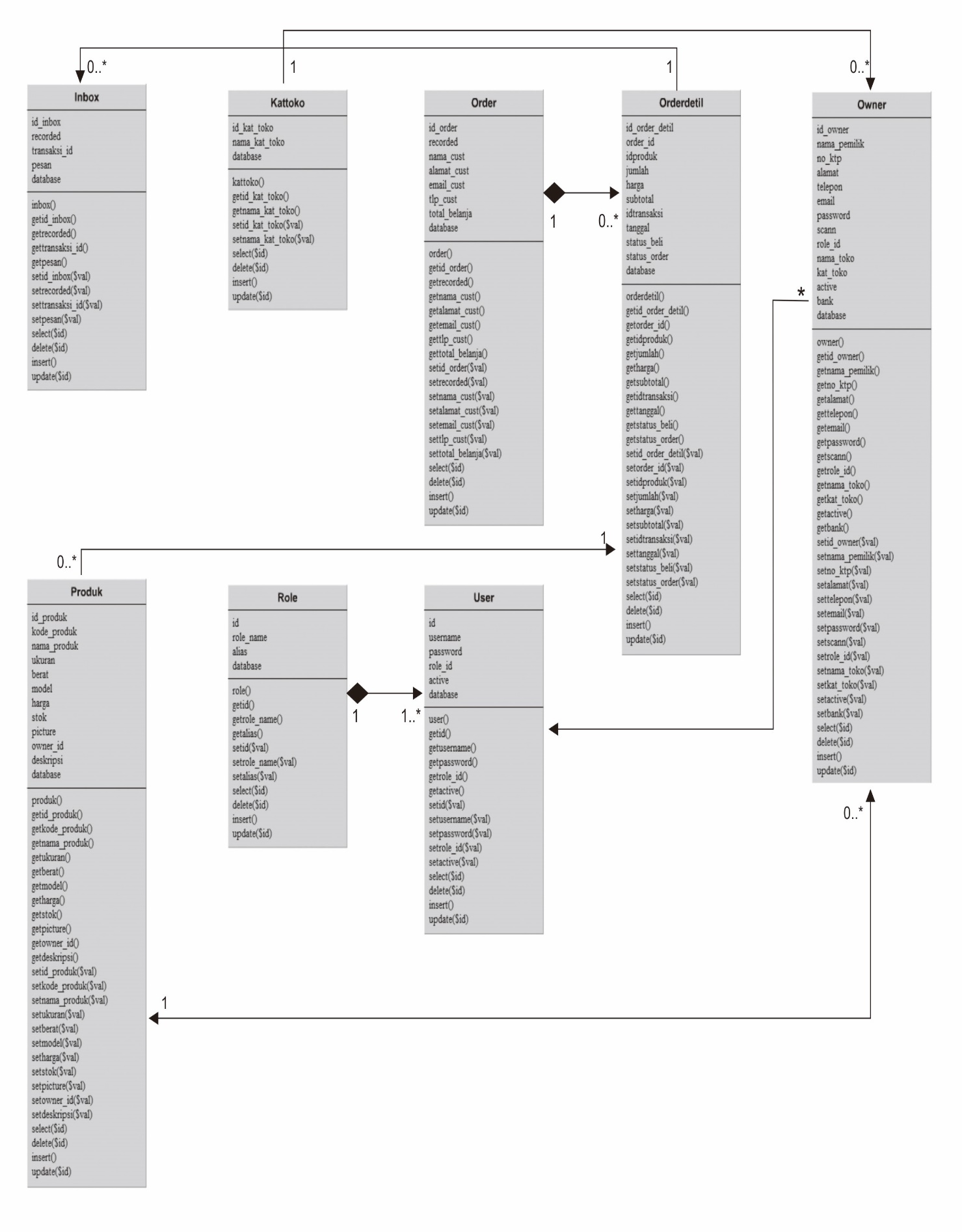
**Gambar 9. *Sequence* Pemesanan Konsumen**

**Keterangan :**

Konsumenmeliha katalog produk, lalu memilih produk, setelah itu konsumen menginputkan jumlah produk yang di pilih. Konsumen dapat melihat belanjaannya dan *checkout*. Saat *checkout* konsumen wajib mengisi *form*pembelian, lalu sistem akan mengambil data pesanan tersebut dan menyimpannya di *database*. Setelah disimpan sistem akan memberi konfirmasi pembelian. Untuk mengecek orderan konsumen memasukan ID transaksi, lalu sistem akan mengecek dan mem*valid*asi id tersebut. Jika *valid*asi sukses, maka sistem akan menampilkan halaman Cek *Order*.

Konsumen dapat memberi komentar dengan meng*input*kan komentar, lalu sistem akan menyimpan komentar tersebut kedalam *database*.

* + - 1. *Class Diagram*

****

**Gambar 10. *Class Diagram***

* + Tabel-tabel

Tabel berfungsi untuk menyimpan data-data tentang objek maupun proses yang dilakukan oleh aplikasi yang tersimpan dalam database.

* + 1. *Tabel Roles*

Nama tabel : roles

Deskripsi : Untuk membedakan user pada saat *Login*.

Primary key : id

**Tabel 1. Roles**

|  |  |  |
| --- | --- | --- |
| NAMA | TIPE | PANJANG |
| id | Auto Number | Long Integer |
| role\_name | Text | 255 |
| alias | Text | 255 |

* + 1. *Tabel Users*

Nama tabel : users

Deskripsi : Untuk menyimpan data-data users yang *Login* pada database market.

Primary key : id

**Tabel 2. Users**

|  |  |  |
| --- | --- | --- |
| NAMA | TIPE | PANJANG |
| Id | AutoNumber | Long Integer |
| *username* | Text | 255 |
| *password* | Text | 255 |
| role\_id | Number | 11 |
| Active | Number | 11 |

* + 1. *Tabel Kategori Toko*

Nama tabel : tbl\_kategori\_toko

Deskripsi : Untuk membedakan setiap toko pada database *market*

Primary key : id\_kat\_toko

**Tabel 3. Kategori Toko**

|  |  |  |
| --- | --- | --- |
| NAMA | TIPE | PANJANG |
| id\_kat\_toko | Auto Number | Long Integer |
| nama\_kat\_toko | Text | 255 |

* + 1. *Table Inbox*

Nama tabel : tbl\_inbox

Deskripsi : Untuk menyimpan komentar dari konsumen.

Primary key : id\_inbox

**Tabel 4. Inbox**

|  |  |  |
| --- | --- | --- |
| NAMA | TIPE | PANJANG |
| id\_inbox | Auto Number | Long Integer |
| recorded | Date/Time |  |
| transaksi\_id | Number | 255 |
| Pesan | Text | 255 |

* + 1. *Tabel Order*

Nama tabel : tbl\_*order*

Deskripsi : Untuk menyimpan data konsumen yang sudah *order* pada database*market.*

Primary key : id

**Tabel 5. *Order***

|  |  |  |
| --- | --- | --- |
| NAMA | TIPE | PANJANG |
| id\_*order* | Auto Number | Long Integer |
| recorded | Date/Time |  |
| nama\_cust | Text | 255 |
| alamat\_cust | Text | 255 |
| *email*\_cust | Text | 255 |
| tlp\_cust | Number | 255 |
| total\_belanja | Number | 11 |

* + 1. *Tabel Order Detil*

Nama tabel : tbl\_*order*\_detil

Deskripsi : Untuk menyimpa data *produk* yang sudah di *order*pada database*market*.

Primary key : id\_*order*\_detil

**Tabel 6. *Order* Detil**

|  |  |  |
| --- | --- | --- |
| NAMA | TIPE | PANJANG |
| id\_*order*\_detil | AutoNumber | Long Integer |
| *order*\_id | Number | 11 |
| id*produk* | Number | 11 |
| jumlah | Number | 11 |
| harga | Number | 11 |
| subtotal | Number | 11 |
| idtransaksi | Number | 255 |
| tanggal | Date |  |
| status\_beli | Text | 255 |
| status\_*order* | Text | 255 |

* + 1. *Tabel Owner*

Nama tabel : tbl\_*Owner*

Deskripsi : Untuk menyimpan data-data *Owner* pada database *market.*

Primary key : id\_*Owner*

**Tabel 7. *Owner***

|  |  |  |
| --- | --- | --- |
| NAMA | TIPE | PANJANG |
| id\_*Owner* | AutoNumber | Long Integer |
| nama\_pemilik | Text | 255 |
| no\_ktp | Number | 255 |
| alamat | Text | 255 |
| telepon | Text | 255 |
| *email* | Text | 255 |
| *password* | Text | 255 |
| scann | Text | 255 |
| role\_id | Number | 255 |
| nama\_toko | Text | 255 |
| kat\_toko | Number | 11 |
| active | Text | 255 |
| *bank* | Number | 255 |

* + 1. *Tabel Produk*

Nama tabel : tbl\_*produk*

Deskripsi : Untuk menyimpan data-data *produk* pada database

*market*.

Primary key : id\_*produk*

**Tabel 8. *Produk***

|  |  |  |
| --- | --- | --- |
| NAMA | TIPE | PANJANG |
| id\_*produk* | AutoNumber | Long Integer |
| kode\_*produk* | Number | 11 |
| nama\_*produk* | Text | 255 |
| ukuran | Text | 255 |
| berat | Number | 11 |
| model | Text | 255 |
| harga | Number | 11 |
| stok | Number | 11 |
| picture | Text | 255 |
| *Owner*\_id | Number | 11 |
| deskripsi | Text | 255 |