

BAB IV

HASIL DAN PEMBAHASAN

4.1 Hasil penelitian

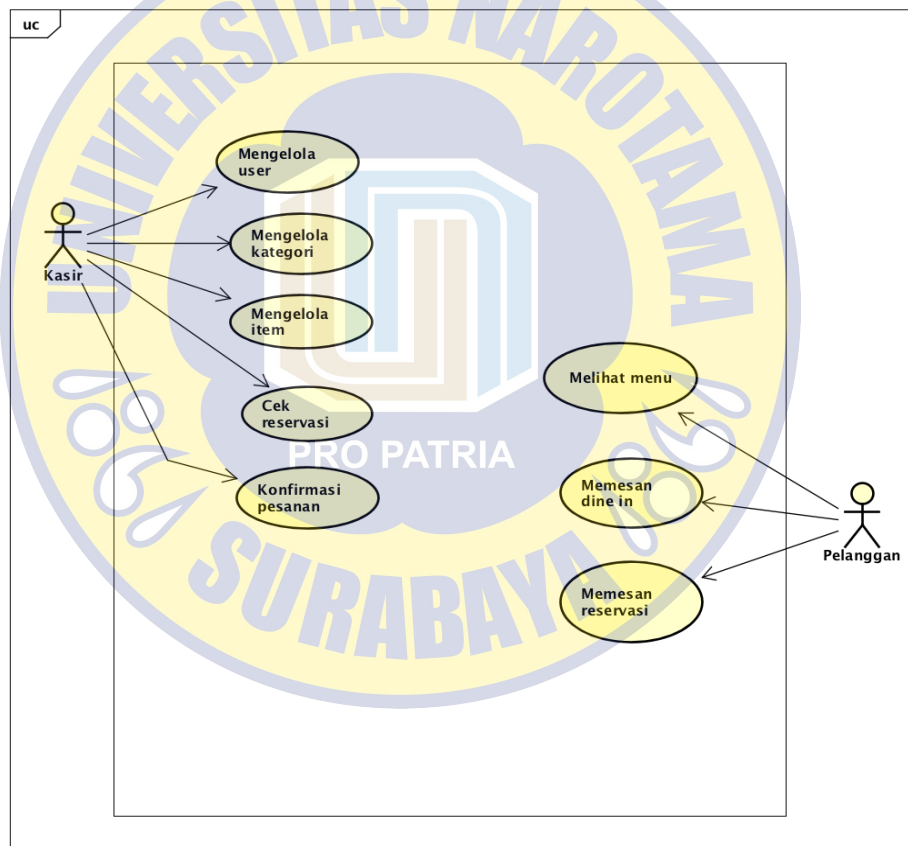
Pada penelitian ini menghasilkan aplikasi yang sesuai dengan kebutuhan pada DE'OAK CAFE. Dalam tahapan desain aplikasi penelitian ini menggunakan USER CENTER DESIGN (UCD) sebagai metodenya. Menu desain aplikasi android yang dirancang menggunakan *bottom navigation*. Sistem yang dibangun memiliki tampilan yang mudah digunakan oleh pengguna (*friendly interface*) [13], sehingga pelanggan tidak perlu pembelajaran khusus terkait aplikasi yang ada. Menu yang ada pada aplikasi android dibuat sederhana dengan fokus permasalahan utamanya yaitu reservasi. Reservasi dibagi menjadi dua yaitu dine in dan reservasi (*package*). Dine in berfungsi untuk reservasi ketika di tempat lokasi DE'OAK CAFE sehingga sudah tidak perlu lagi memanggil waiters maupun antri pada kasir, karena pemesanan sudah bisa dilakukan melalui aplikasi android.

Pemakaian aplikasi desktop berbasis web yang sudah di implementasikan dapat berfungsi dengan baik. Eksperimental mencakup perancangan prototipe berdasarkan perubahan yang dilakukan[9], dengan adanya sistem yang terotomasi aplikasi android dengan *back office*, kinerja karyawan semakin mudah dan pelanggan tidak perlu lagi menelpon kepada DE'OAK CAFE untuk memastikan reservasi. Kasir sudah tidak menginputkan lagi detail pesanan dan data pelanggan yang sudah melakukan reservasi. Dengan adanya kode barcode, kasir sudah dapat mengetahui detail reservasi pelanggan.

4.2 Unified Modelling Language (UML)

Pada sub bab ini akan dijelaskan mengenai UML program. Penggunaan UML pada penelitian ini menjelaskan tentang keseluruhan program dalam menyajikan desain dan alur aplikasi. Peneliti juga menyajikan kasus penggunaan dan fitur yang ditambahkan ke aplikasi[8]. Berikut penjelasan mengenai UML pada penelitian ini.

4.2.1 Use Case



Gambar 4.1 Use Case

4.1 Tabel Use Case

No. UC	Nama Use Case	Aktor	Keterangan
1	Mengelola user	Kasir	-
2	Mengedit kategori	Kasir	-
3	Menghapus item	Kasir	-
4	Cek reservasi	Kasir	-
5	Konfirmasi pesanan	Kasir	-
6	Melihat menu	Pelanggan	-
7	Memesan dine in	Pelanggan	-
8	Memesan reservasi	Pelanggan	-

Penjelasan Use Case :

Use Case Mengelola User

Use Case Name	Mengelola User
Actor	Kasir
Description	Use Case ini menjelaskan tentang mengelola user
Trigger	Kasir melakukan klik menu User
Relationship :	
➤ Association	Kasir
➤ Include	Log In
➤ Extend	List User, Mengedit User, Meghapus User
➤ Generation	-
Pre-Condition	Komputer menyala dan sistem normal
Normal Flow of Event	1. Sistem menampilkan form.M. halaman utama 2. Kasir memilih menu User 3. Sistem menampilkan form"N". Halaman

	Modul User
Subflow	-
Exceptional Flow	-
Post-Condition	-

Use Case Mengelola Kategori

Use Case Name	Mengelola Kategori
Actor	Kasir
Description	Use Case ini menjelaskan tentang mengelola kategori
Trigger	Kasir melakukan klik menu Kategori
Relationship :	➤ Association Kasir ➤ Include Log In ➤ Extend List Kategori, Mengedit Kategori, Meghapus Kategori ➤ Generation -
Pre-Condition	Komputer menyala dan sistem normal
Normal Flow of Event	1. Sistem menampilkan form.M. halaman utama 2. Kasir memilih menu Kategori 3. Sistem menampilkan form"N". Halaman Modul Kategori
Subflow	-
Exceptional Flow	-
Post-Condition	-

Use Case Mengelola Item

Use Case Name	Mengelola Item
Actor	Kasir
Description	Use Case ini menjelaskan tentang mengelola item
Trigger	Kasir melakukan klik menu Item
Relationship :	➤ Association Kasir ➤ Include Log In ➤ Extend List Item, Mengedit Item, Meghapus Item ➤ Generation -
Pre-Condition	Komputer menyala dan sistem normal
Normal Flow of Event	1. Sistem menampilkan form.M. halaman utama 2. Kasir memilih menu Item 3. Sistem menampilkan form"N". Halaman Modul Item
Subflow	-
Exceptional Flow	-
Post-Condition	-

Use Case Cek Reservasi

Use Case Name	Cek reservasi
Actor	Kasir
Description	Use Case ini menjelaskan tentang cek reservasi
Trigger	Kasir melakukan klik list reservasi
Relationship :	➤ Association Kasir ➤ Include Log In

➤ Extend	Mengedit Item, Meghapus Item
➤ Generation	-
Pre-Condition	Komputer menyala dan sistem normal
Normal Flow of Event	1. Sistem menampilkan form.M. halaman pesanan 2. Kasir memilih menu List Reservasi 3. Sistem menampilkan form”N”. Halaman Modul Item
Subflow	-
Exceptional Flow	-
Post-Condition	-

Use Case Konfirmasi Pesanan

Use Case Name	Konfirmasi Pesanan
Actor	Kasir
Description	Use Case ini menjelaskan tentang konfirmasi pesanan
Trigger	Kasir melakukan klik cari
Relationship :	
➤ Association	Kasir
➤ Include	Log In
➤ Extend	-
➤ Generation	
Pre-Condition	Komputer menyala dan sistem normal
Normal Flow of Event	1. Sistem menampilkan form.M. halaman pesanan 2. Kasir menginput atau scan kode barcode.

	Sistem menampilkan detail pesanan. 3. Kasir klik simpan
Subflow	-
Exceptional Flow	-
Post-Condition	- Data perubahan sudah di proses, data perubahan tersimpan

Use Case Melihat menu

Use Case Name	Melihat menu
Actor	Pelanggan
Description	Use Case ini menjelaskan tentang melihat menu
Trigger	Pelanggan melakukan klik menu Home
Relationship :	➤ Association Pelanggan ➤ Include Log In ➤ Extend - ➤ Generation -
Pre-Condition	Smartphone menyala dengan normal
Normal Flow of Event	1. Sistem menampilkan form.M. halaman utama. 2. Pelanggan memilih menu Home. 3. Sistem menampilkan form"N". Halaman Modul Home.
Subflow	-
Exceptional Flow	-
Post-Condition	-

Use Case Memesan Dine In

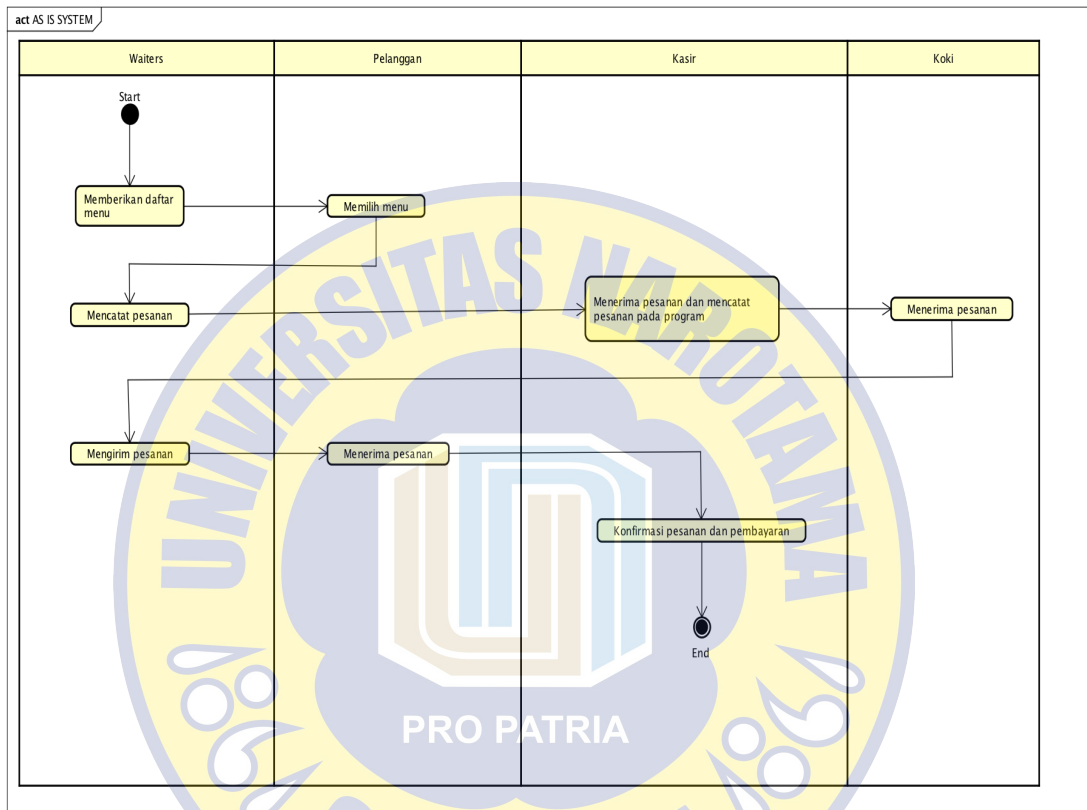
Use Case Name	Memesan Dine in
Actor	Pelanggan
Description	Use Case ini menjelaskan tentang memesan dine in
Trigger	Pelanggan melakukan klik tombol pesan
Relationship :	➤ Association Pelanggan ➤ Include Log In ➤ Extend - ➤ Generation -
Pre-Condition	Smartphone menyala dengan normal
Normal Flow of Event	4. Sistem menampilkan form.M. halaman utama. 5. Pelanggan memilih menu Dine In. 6. Sistem menampilkan form”N”. Halaman Modul Dine In. 7. Pelanggan klik tombol pesan. 8. Pelanggan klik tombol pesan “Sistem menampilkan info”. 9. Pelanggan klik menu pesanan. 10. Pelanggan klik tombol cek pesanan hari ini “Sistem menampilkan detail pesanan” 11. Pelanggan klik tombol simpan “Sistem menyimpan data dan menampilkan barcode”.
Subflow	-
Exceptional Flow	-
Post-Condition	- Data perubahan sudah di proses, data perubahan tersimpan

Use Case Memesan Reservasi

Use Case Name	Memesan Reservasi
Actor	Pelanggan
Description	Use Case ini menjelaskan tentang memesan reservasi
Trigger	Pelanggan melakukan klik tombol pesan
Relationship :	➤ Association Pelanggan ➤ Include Log In ➤ Extend - ➤ Generation -
Pre-Condition	Smartphone menyala dengan normal
Normal Flow of Event	1. Sistem menampilkan form.M. halaman utama. 2. Pelanggan memilih menu Reservasi. 3. Sistem menampilkan form”N”. Halaman Modul Reservasi. 4. Pelanggan klik tombol pesan. 5. Pelanggan klik tombol pesan “Sistem menampilkan info”. 6. Pelanggan klik menu pesanan. 7. Pelanggan klik tombol cek reservasi “Sistem menampilkan detail pesanan dan input kelengkapan reservasi” 8. Pelanggan klik tombol simpan “Sistem menyimpan data dan menampilkan barcode”.
Subflow	-
Exceptional Flow	-
Post-Condition	- Data perubahan sudah di proses, data perubahan tersimpan

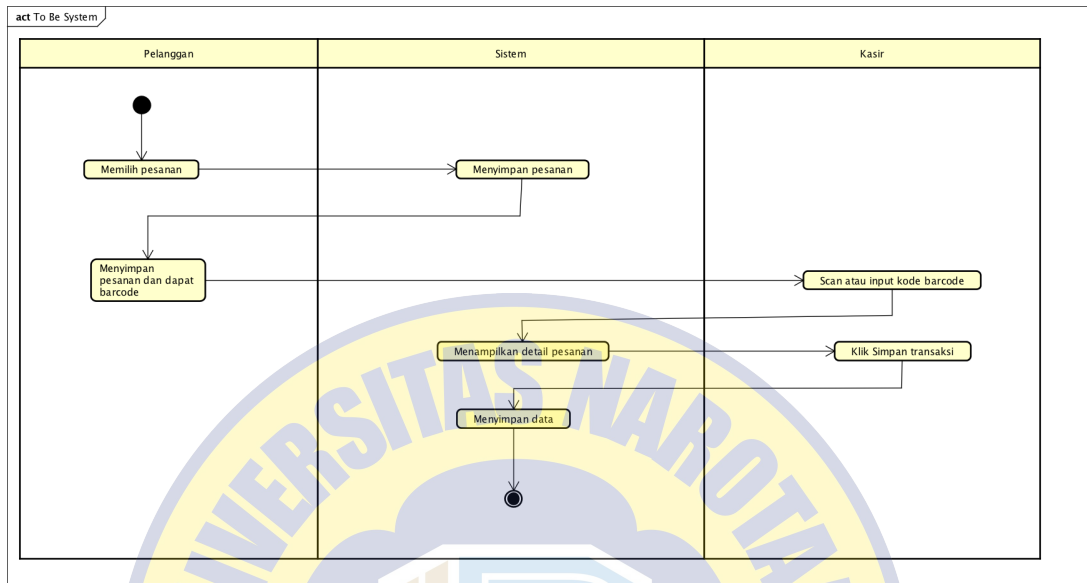
4.2.3 As Is System

Pada sub bab ini menjelaskan tentang alur proses bisnis yang terjadi pada De'Oak Cafe :



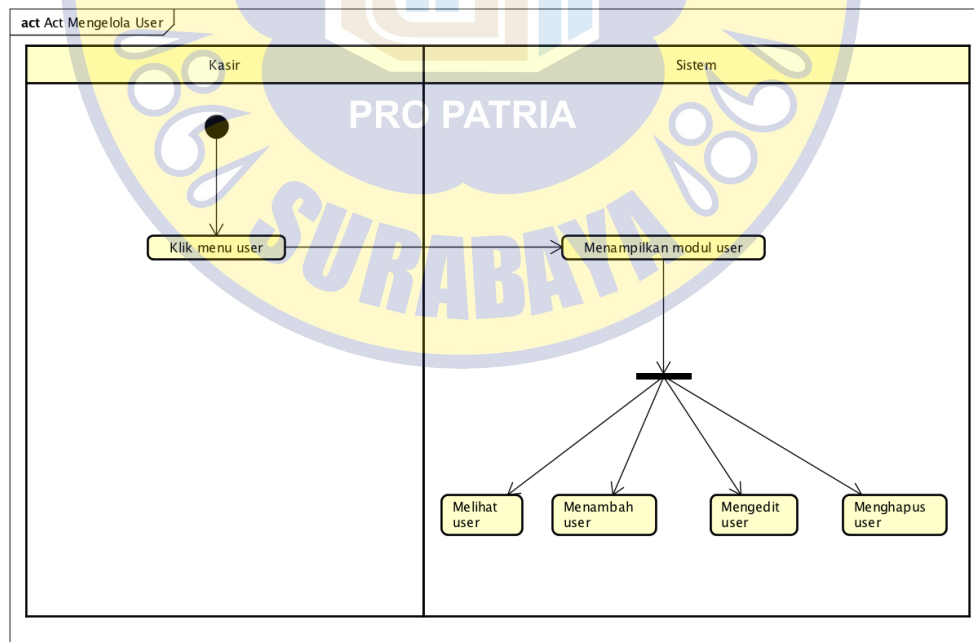
Gambar 4.2 As Is System

4.2.4 To be System

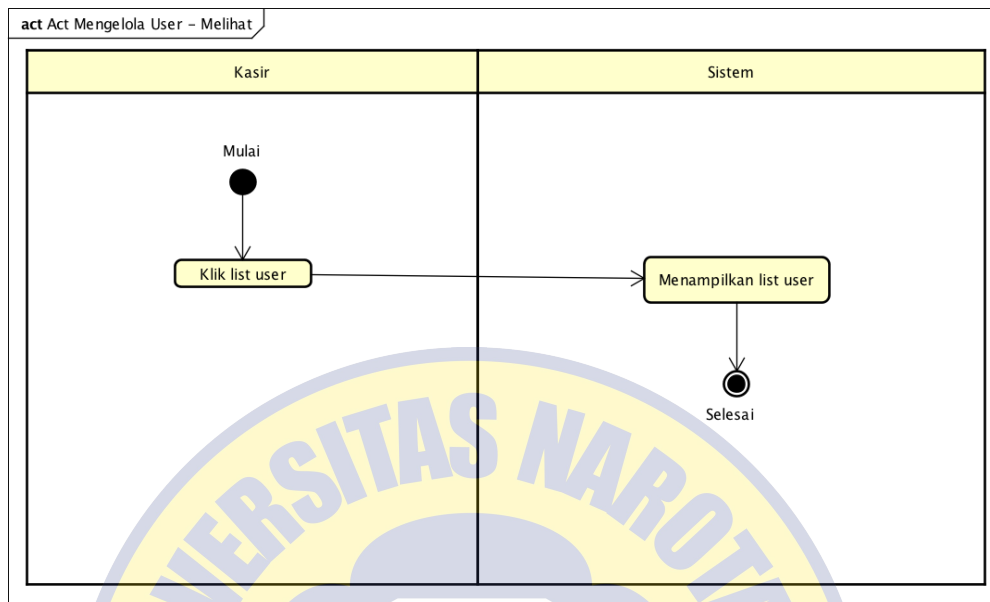


Gambar 4.3 To be System

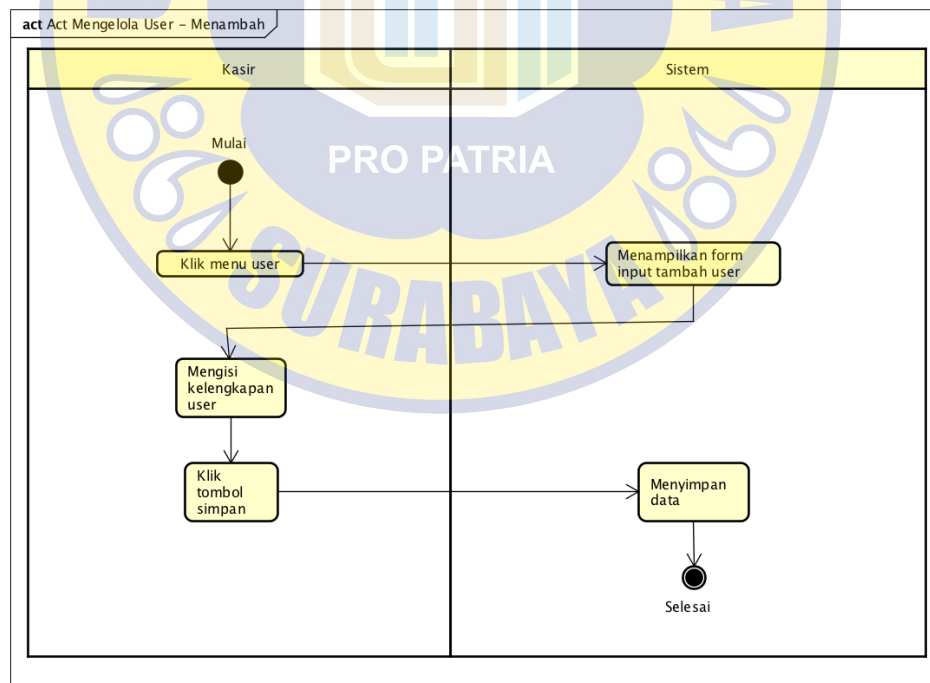
4.2.5 Activity Digram



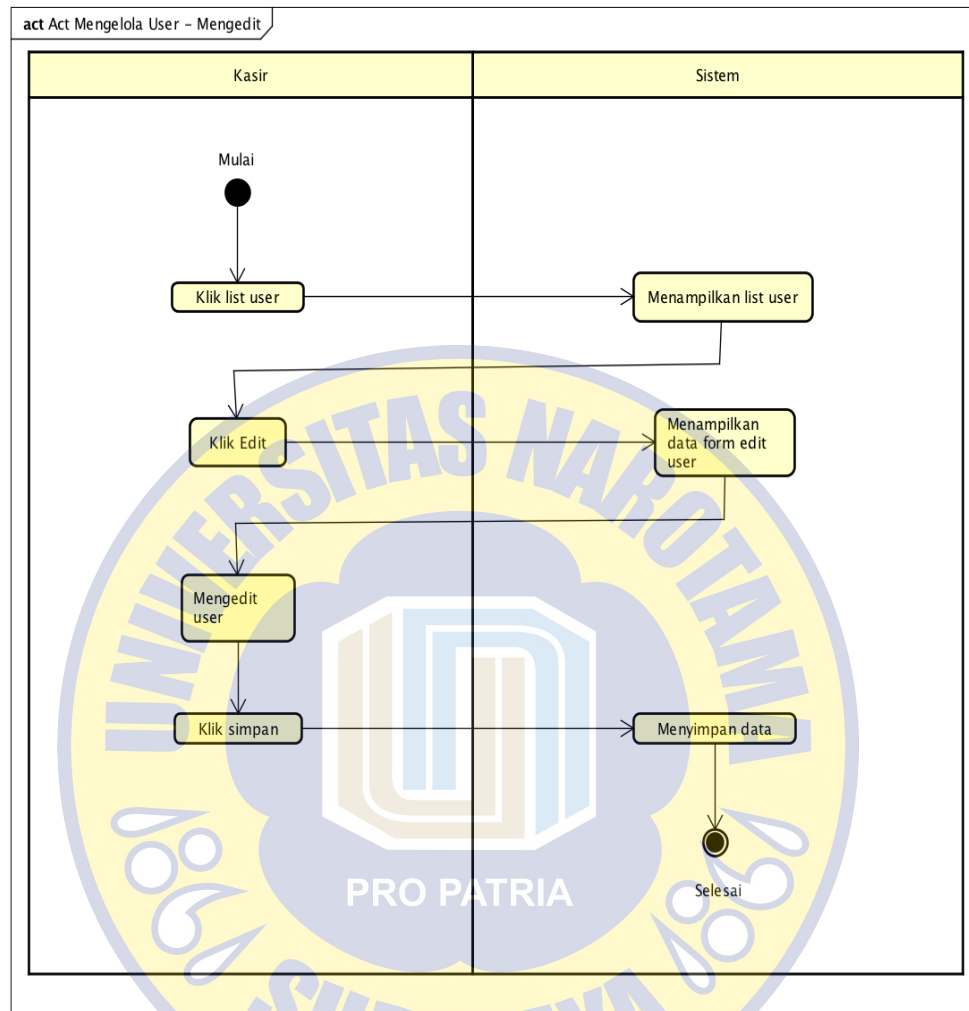
Gambar 4.4 Mengelola User



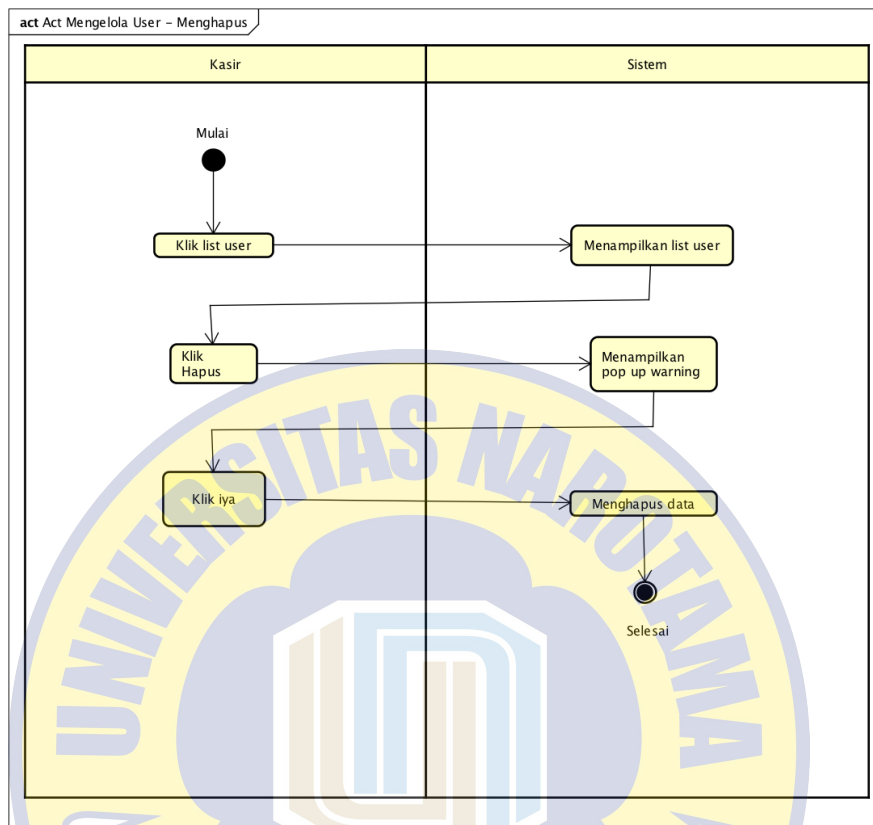
Gambar 4.5 Mengelola User – Melihat



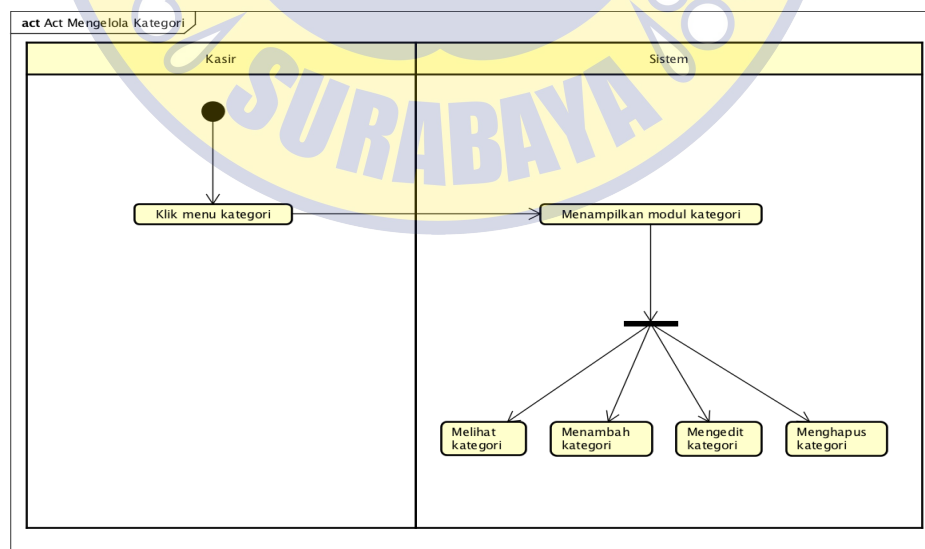
Gambar 4.6 Mengelola User – Menambah



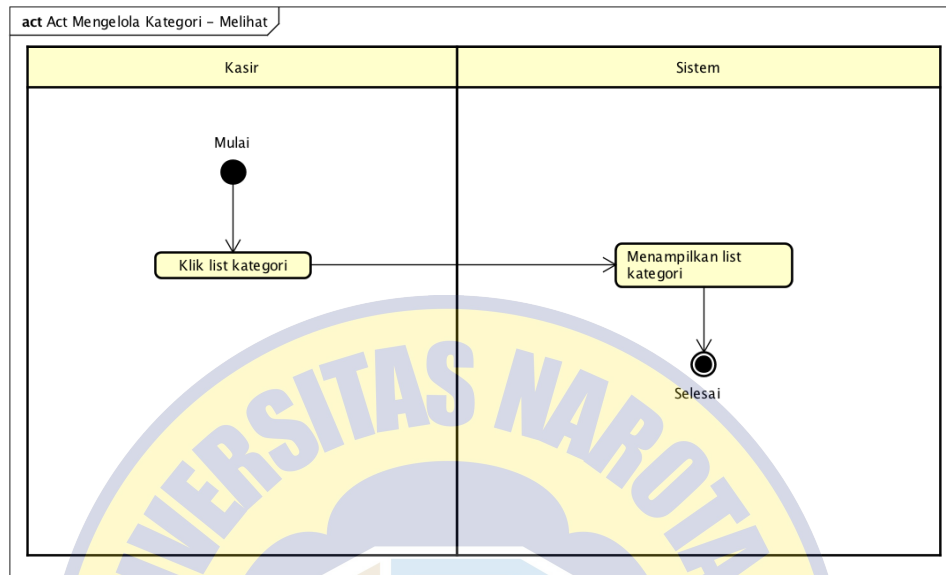
Gambar 4.7 Mengelola User – Mengedit



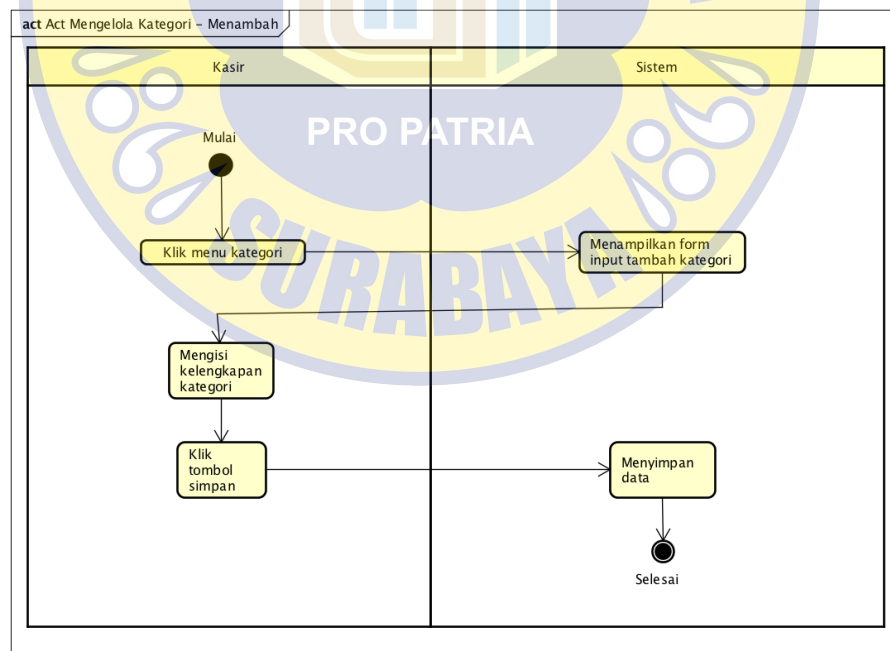
Gambar 4.8 Mengelola User – Menghapus



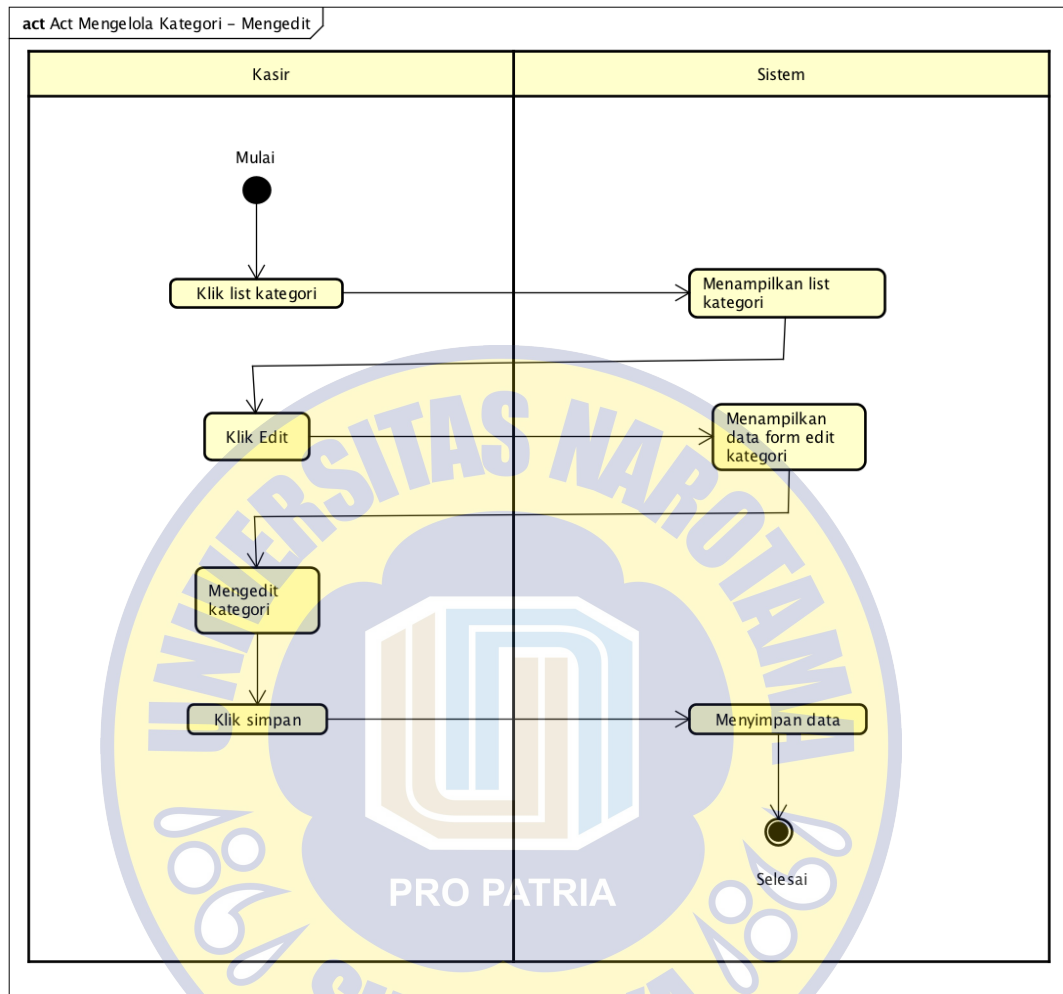
Gambar 4.9 Mengelola Kategori



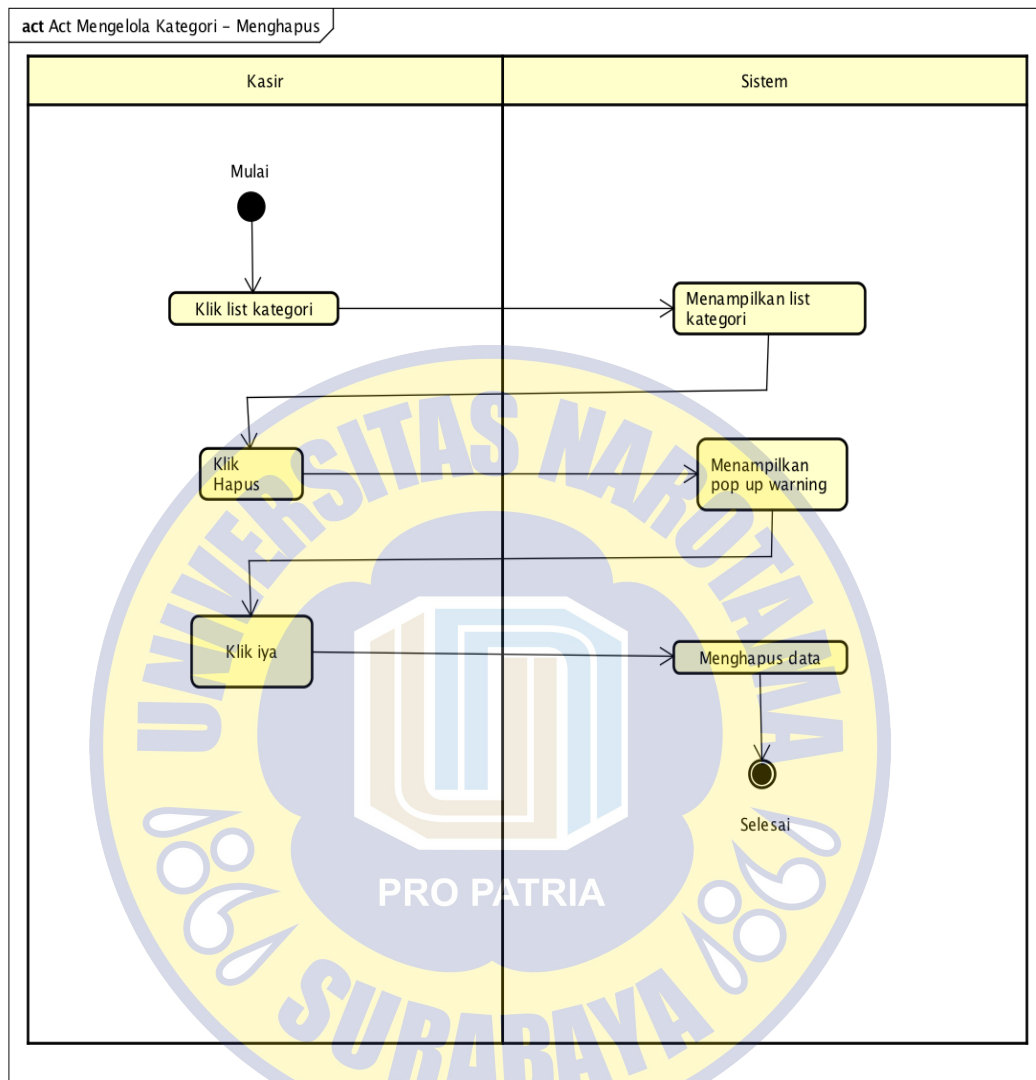
Gambar 4.10 Mengelola Kategori - Melihat



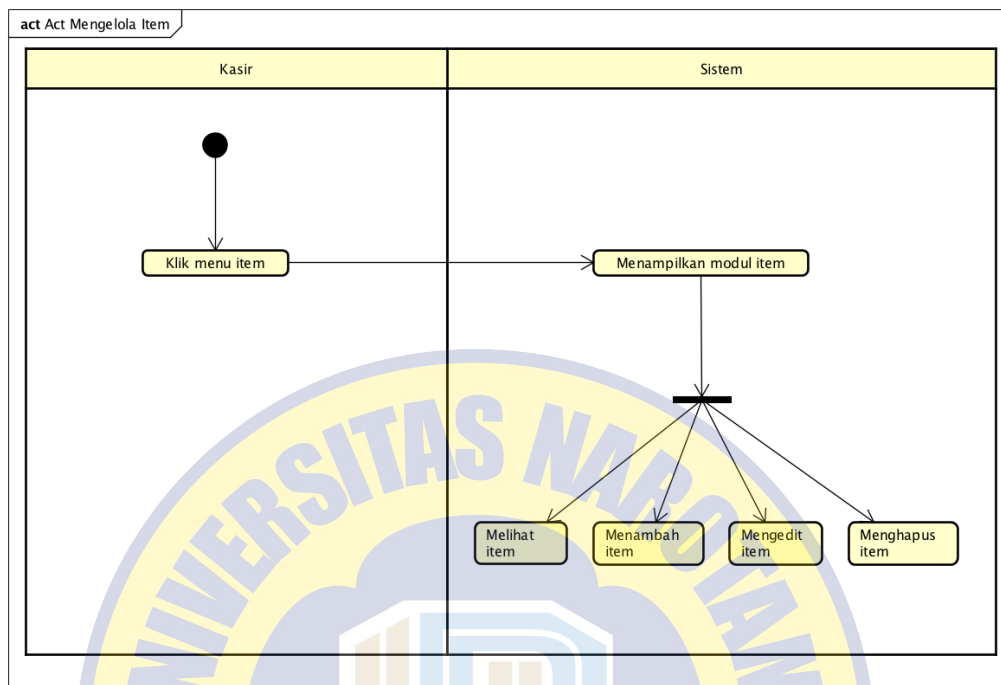
Gambar 4.11 Mengelola Kategori - Menambah



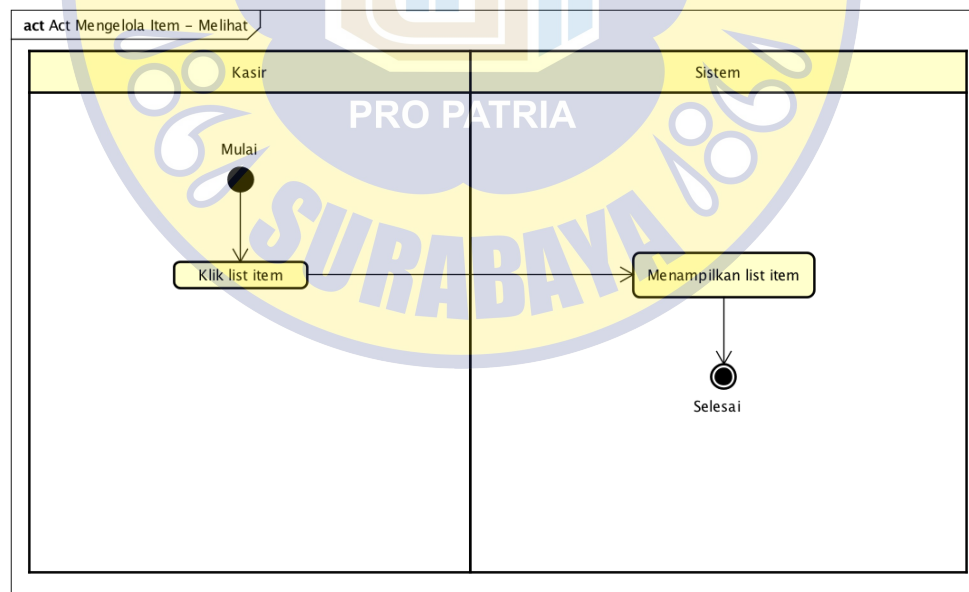
Gambar 4.12 Mengelola Kategori - Mengedit



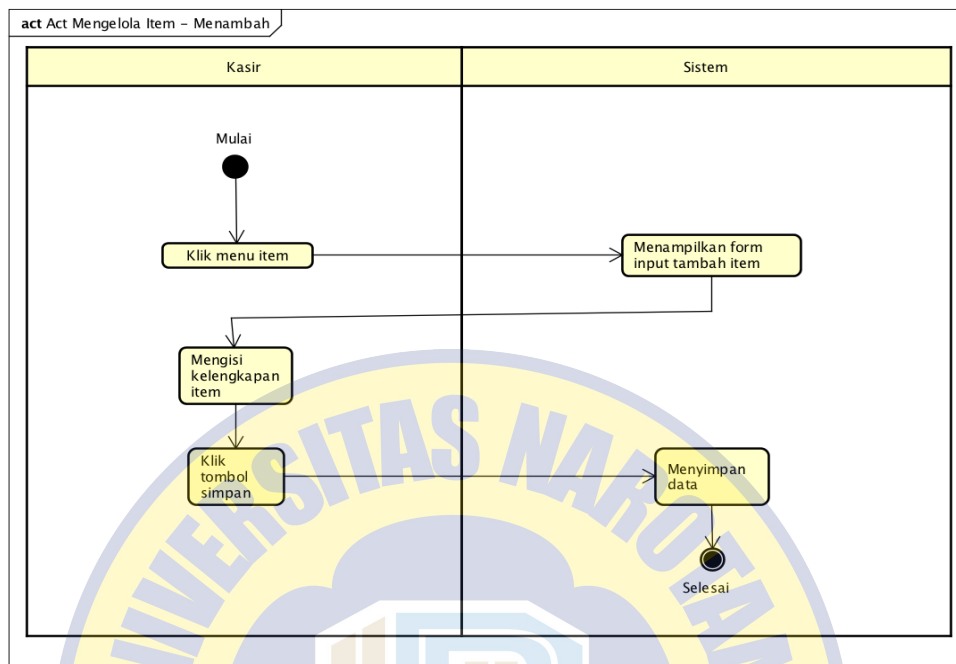
Gambar 4.13 Mengelola Kategori – Menghapus



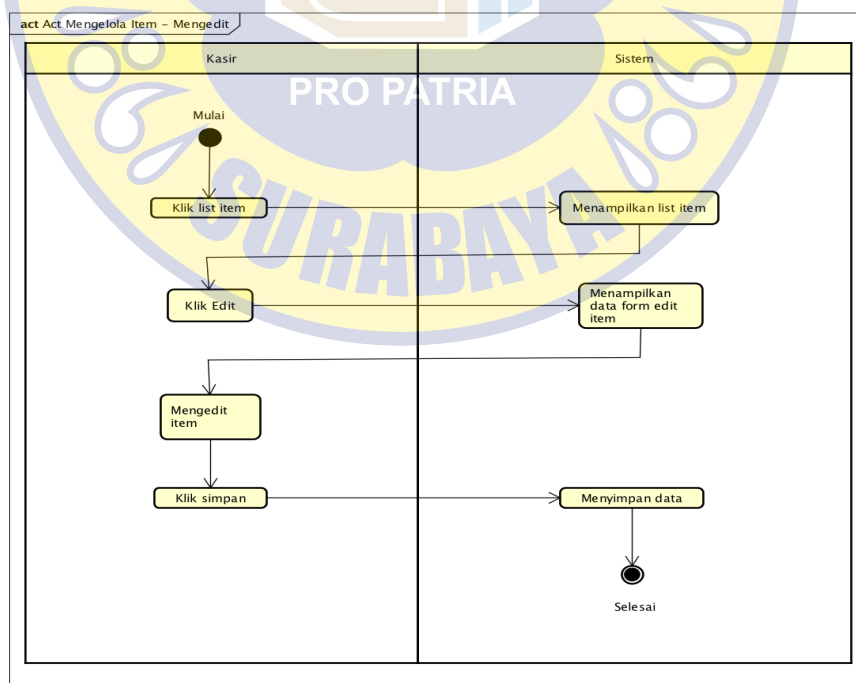
Gambar 4.14 Mengelola Item



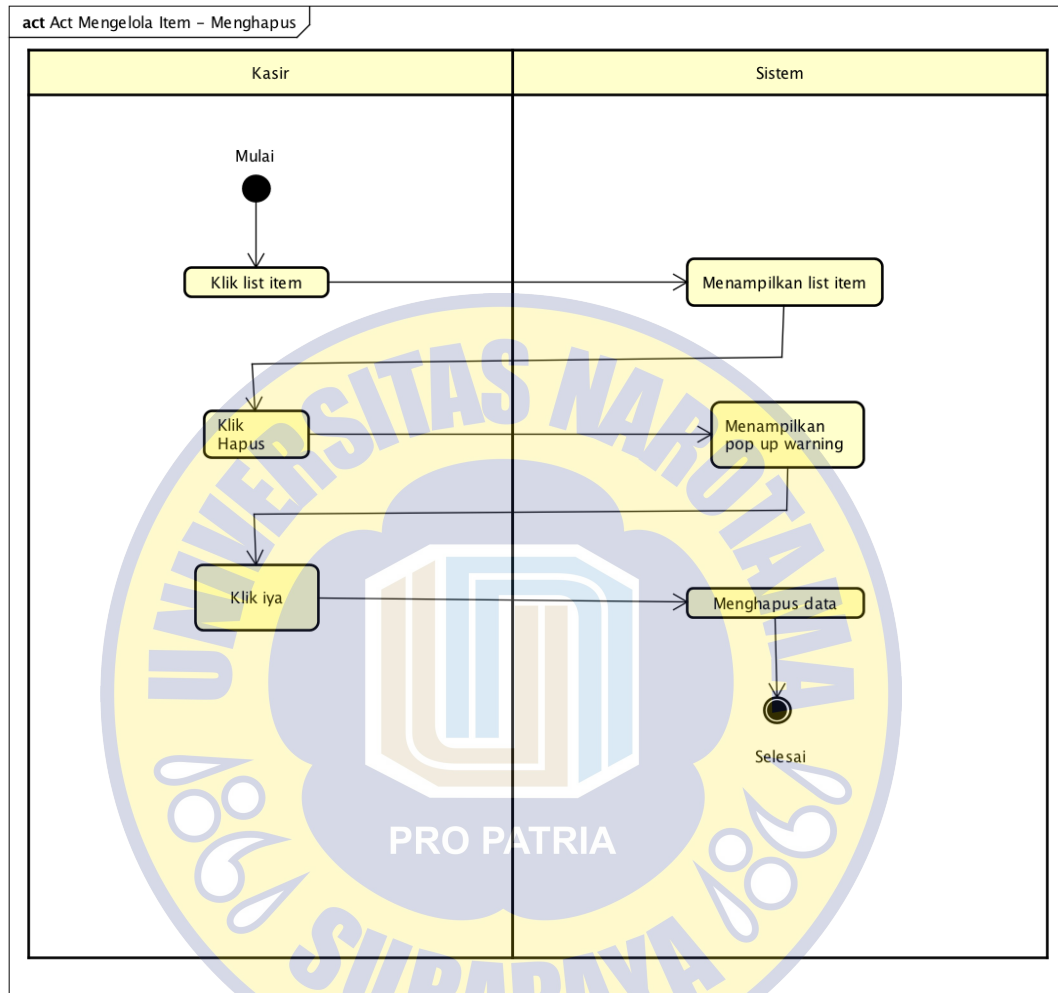
Gambar 4.15 Mengelola Item – Melihat



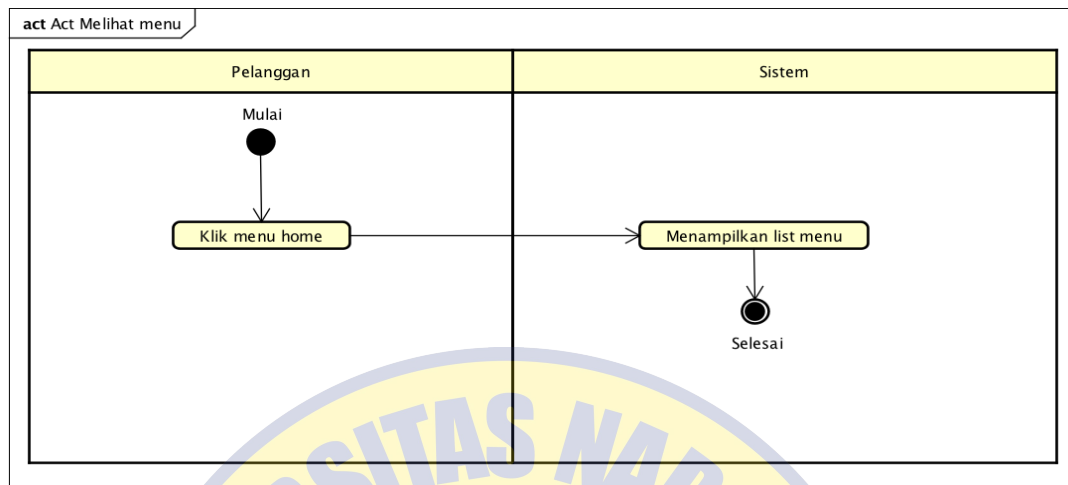
Gambar 4.16 Mengelola Item – Menambah



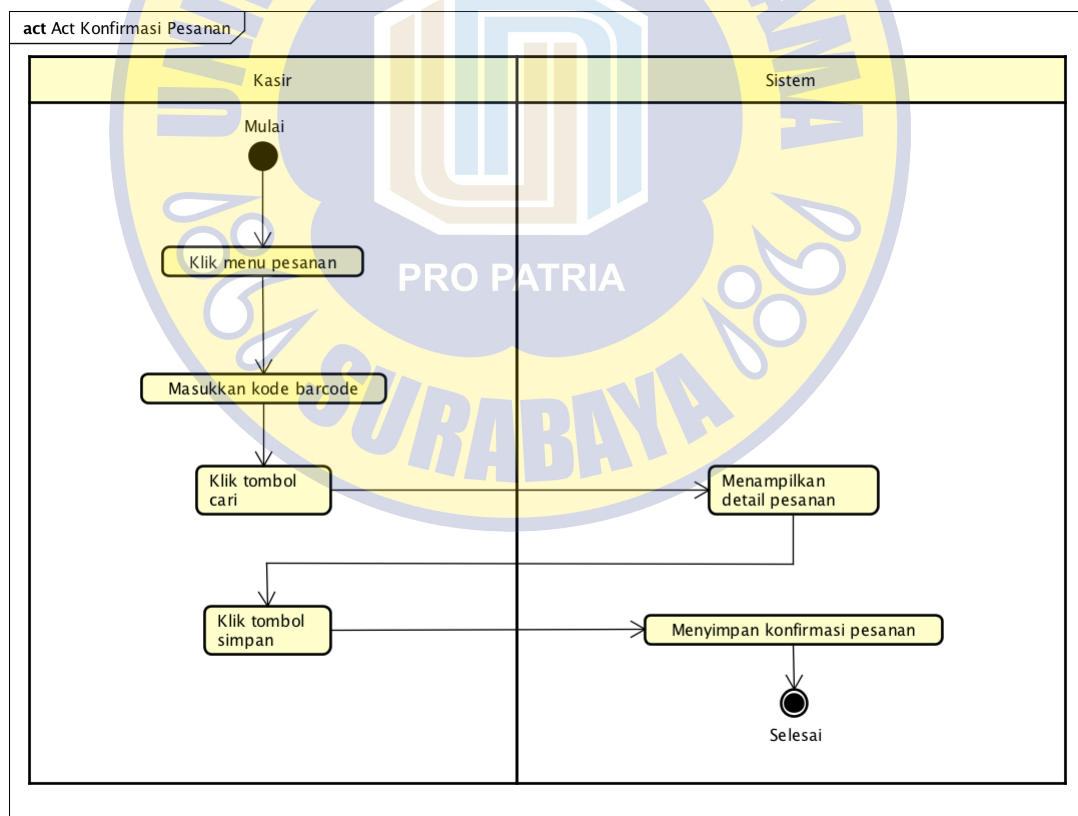
Gambar 4.17 Mengelola Item – Mengedit



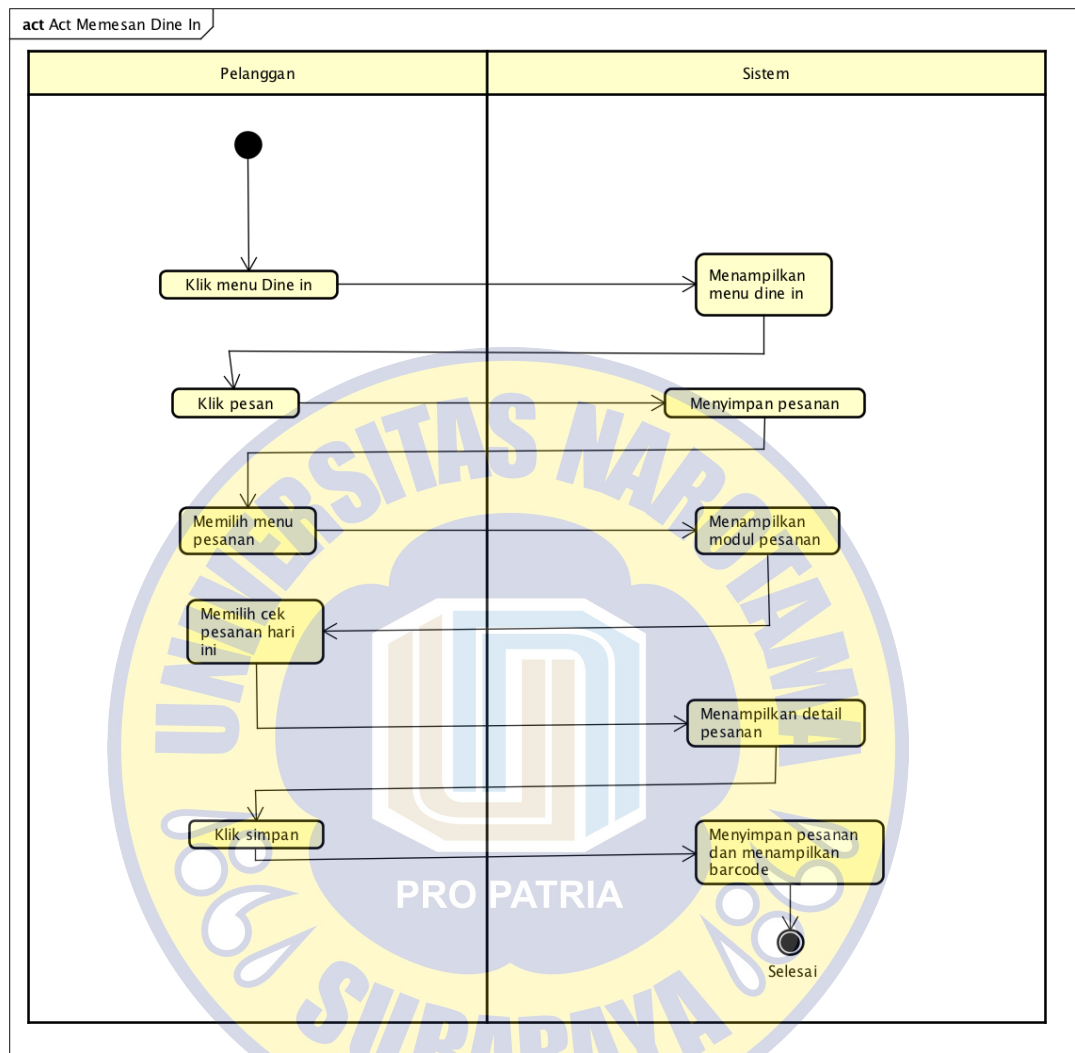
Gambar 4.18 Mengelola Item – Menghapus



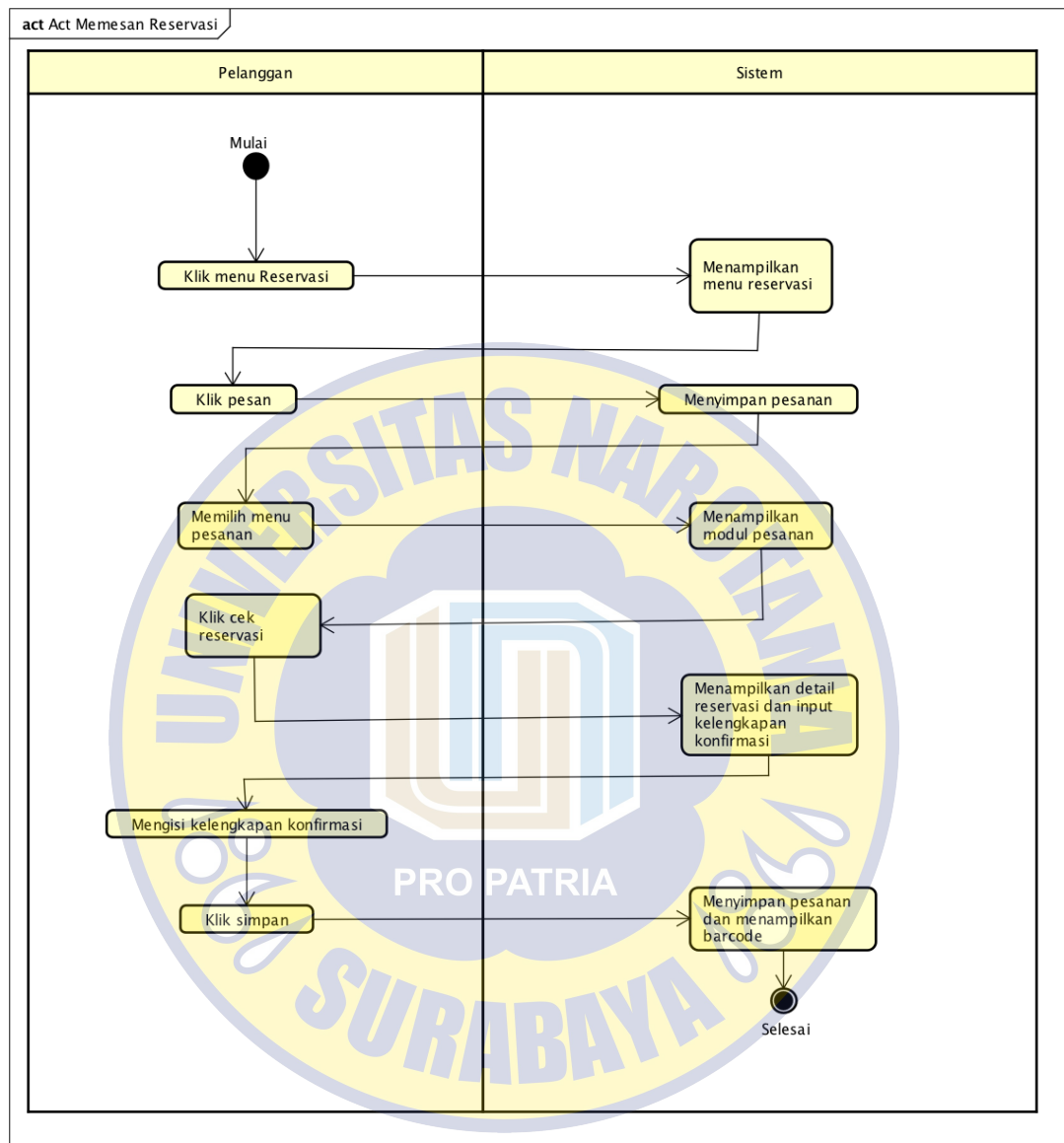
Gambar 4.19 Mengelola melihat menu



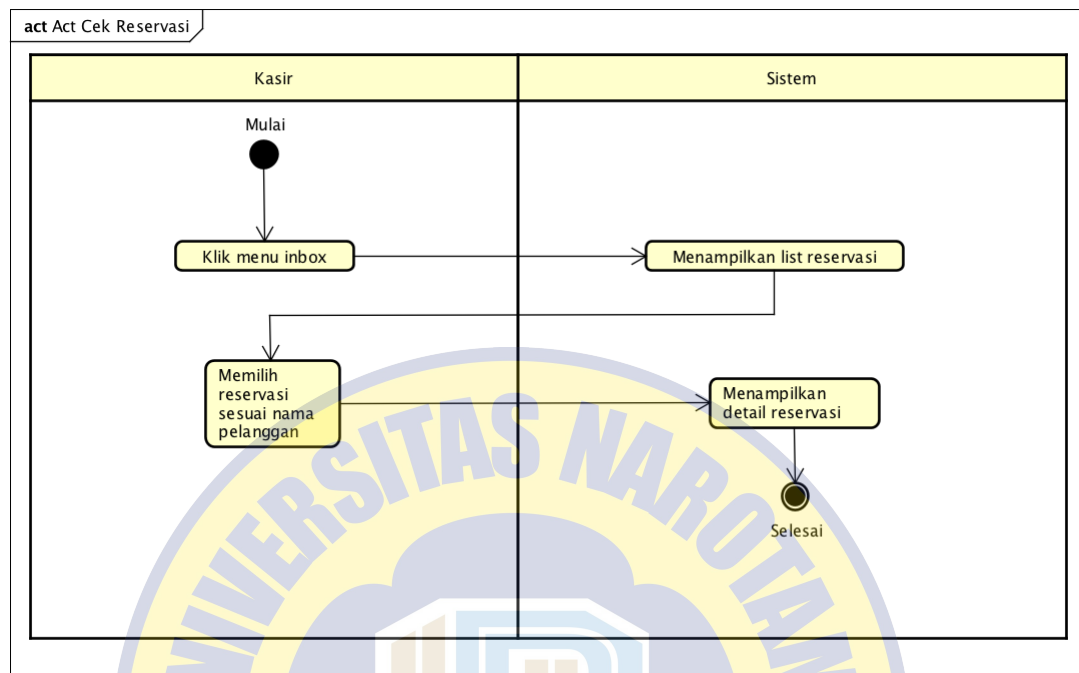
Gambar 4.20 Konfirmasi pesanan



Gambar 4.21 Memesan Dine In

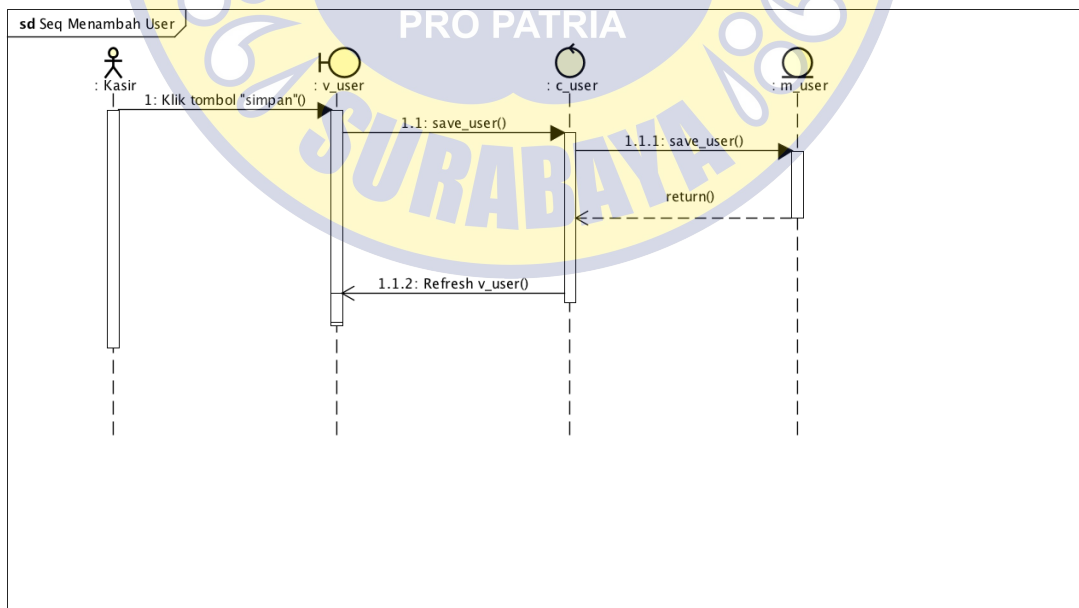


Gambar 4.22 Memesan Reservasi

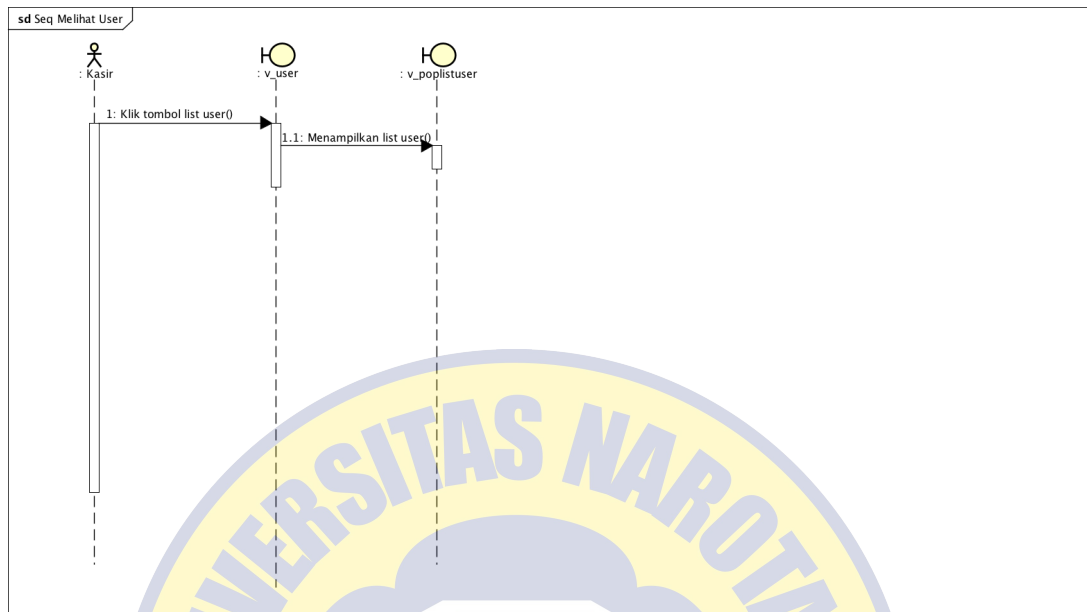


Gambar 4.23 Cek Reservasi

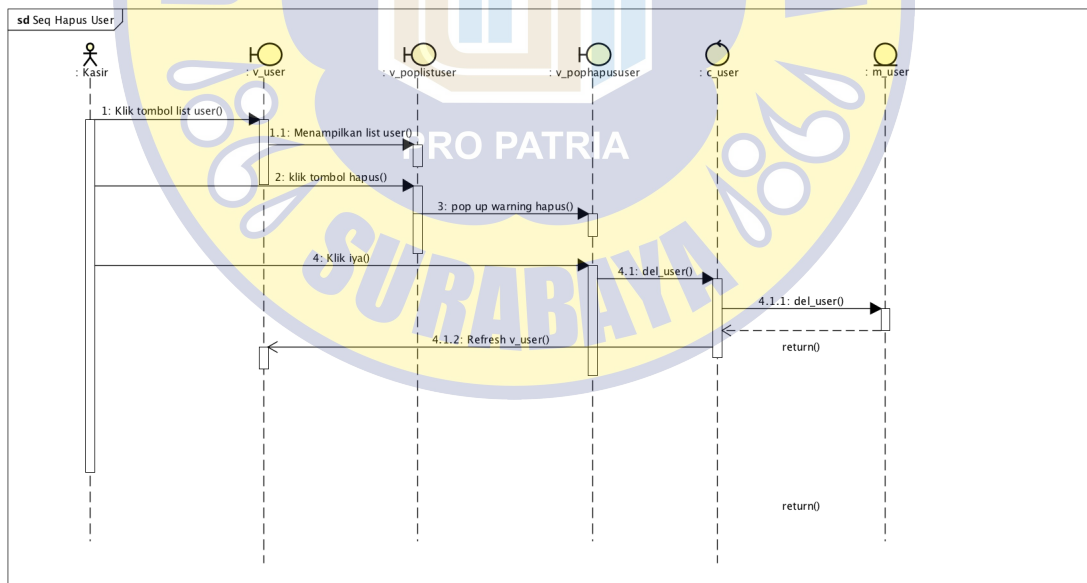
4.2.6 Sequence Diagram



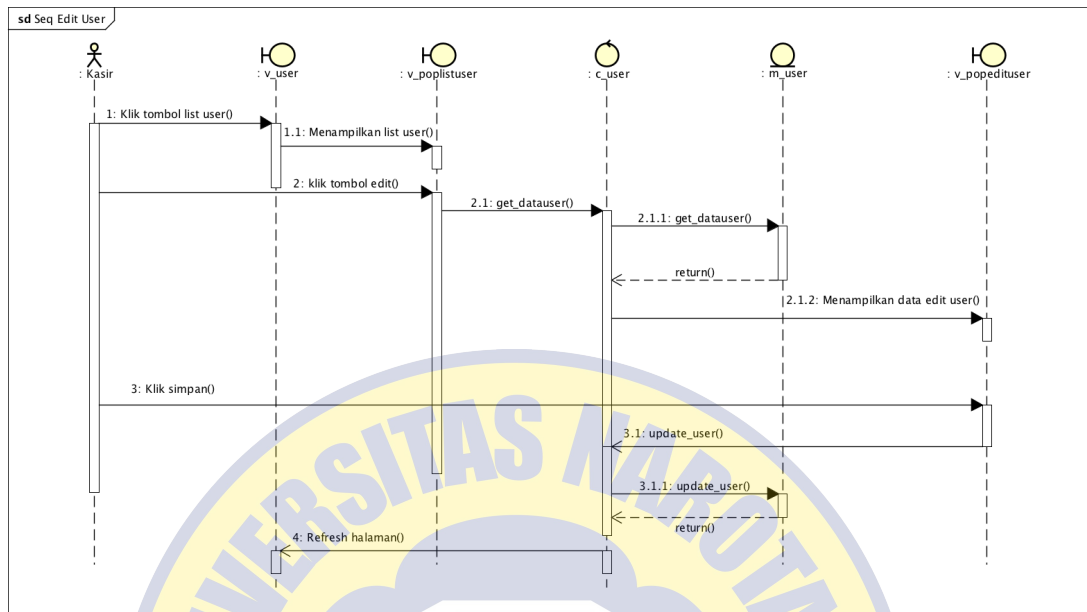
Gambar 4.24 Menambah User



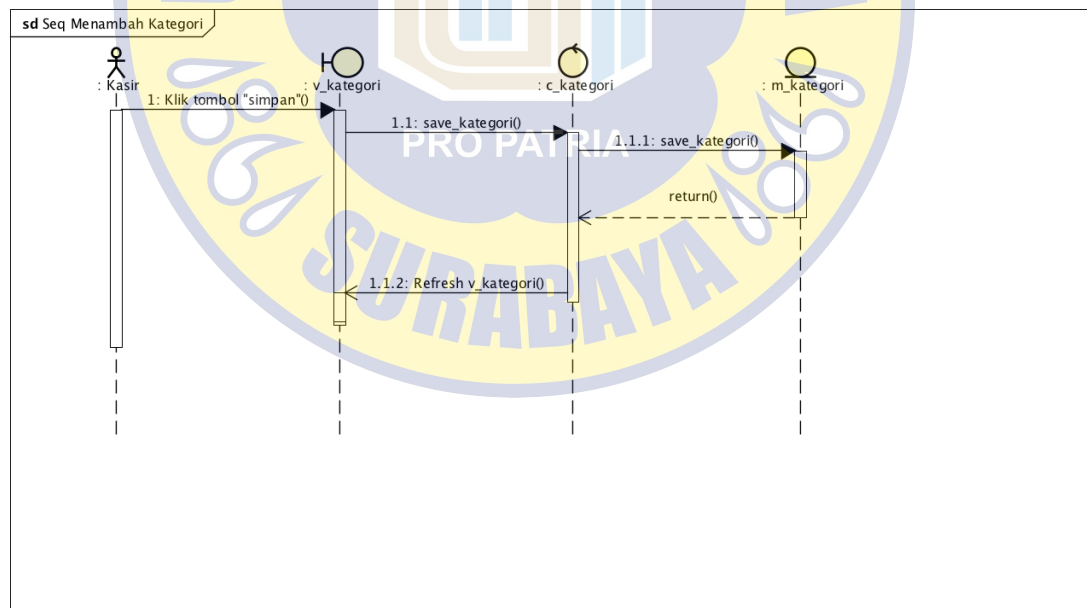
Gambar 4.25 Melihat User



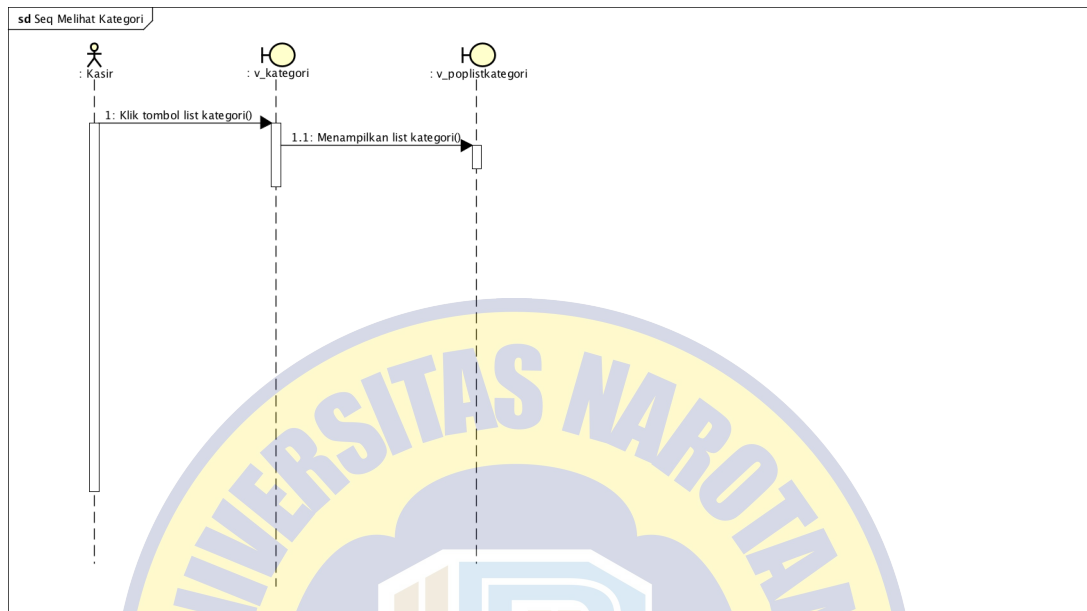
Gambar 4.26 Menghapus user



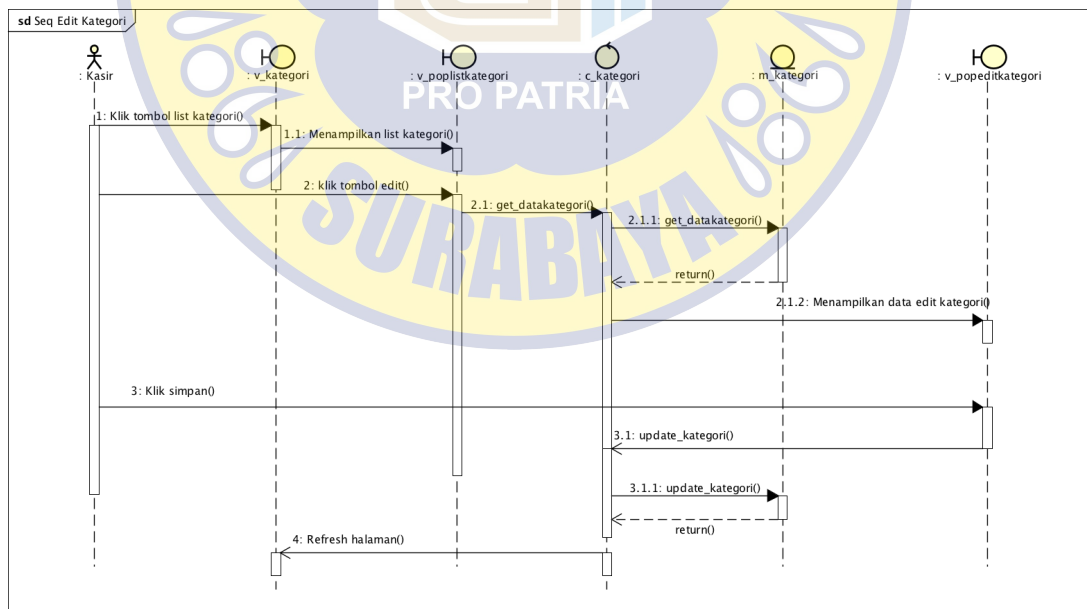
Gambar 4.27 Mengedit user



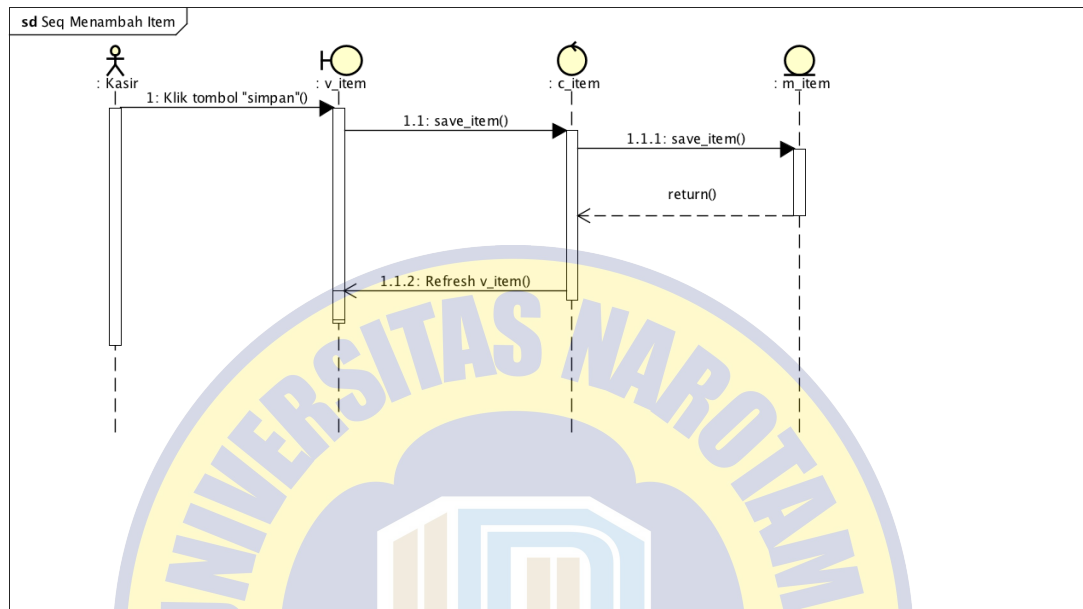
Gambar 4.28 Menambah kategori



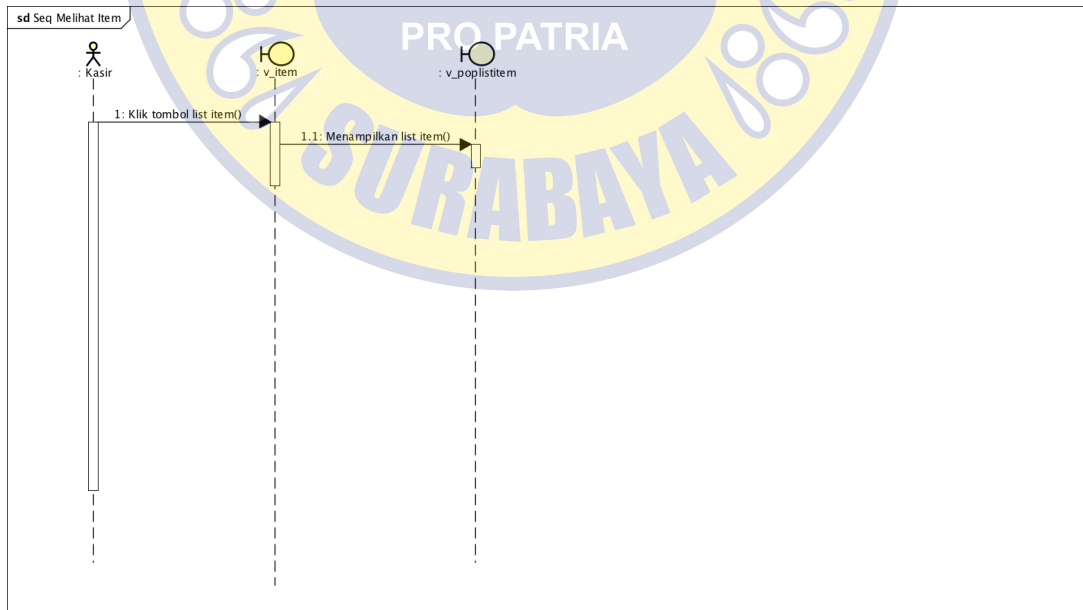
Gambar 4.29 Melihat kategori



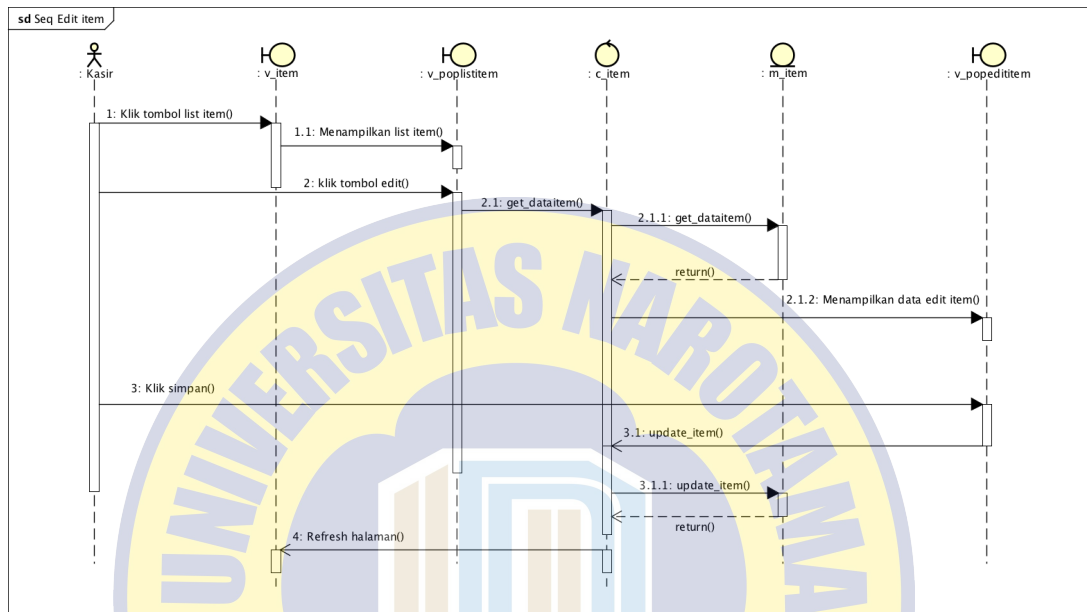
Gambar 4.30 Mengedit kategori



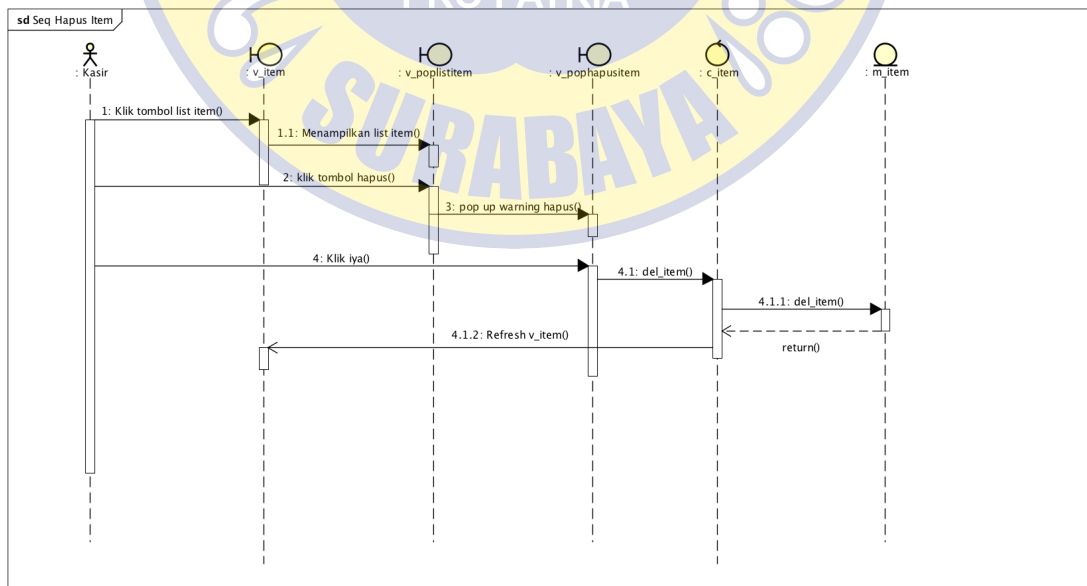
Gambar 4.31 Menambah Item



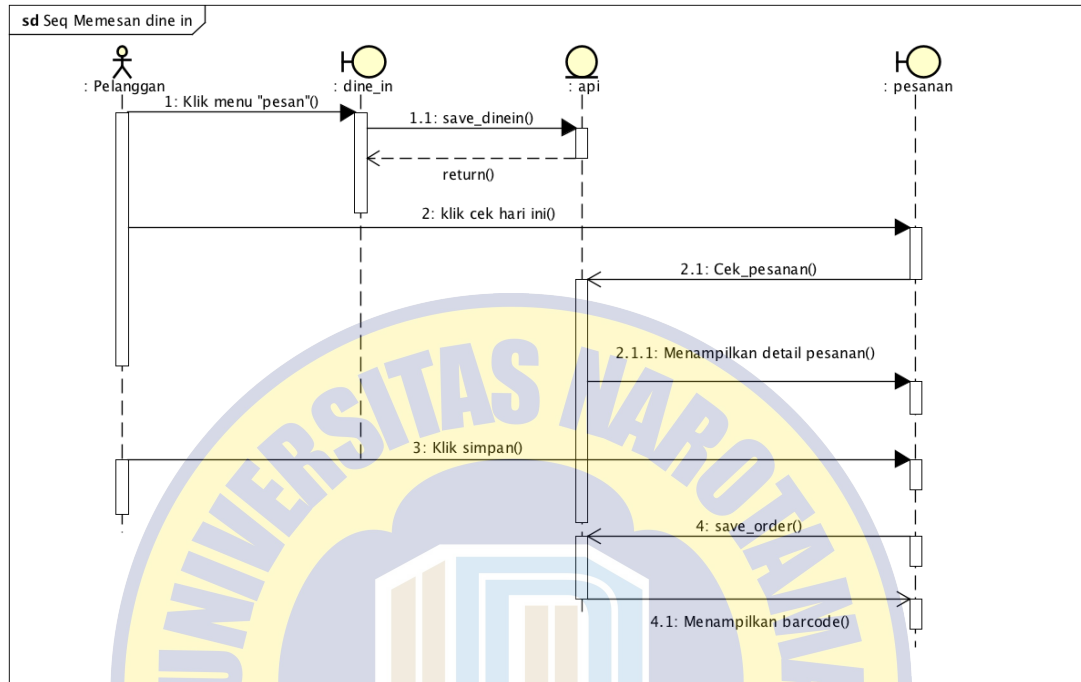
Gambar 4.32 Melihat Item



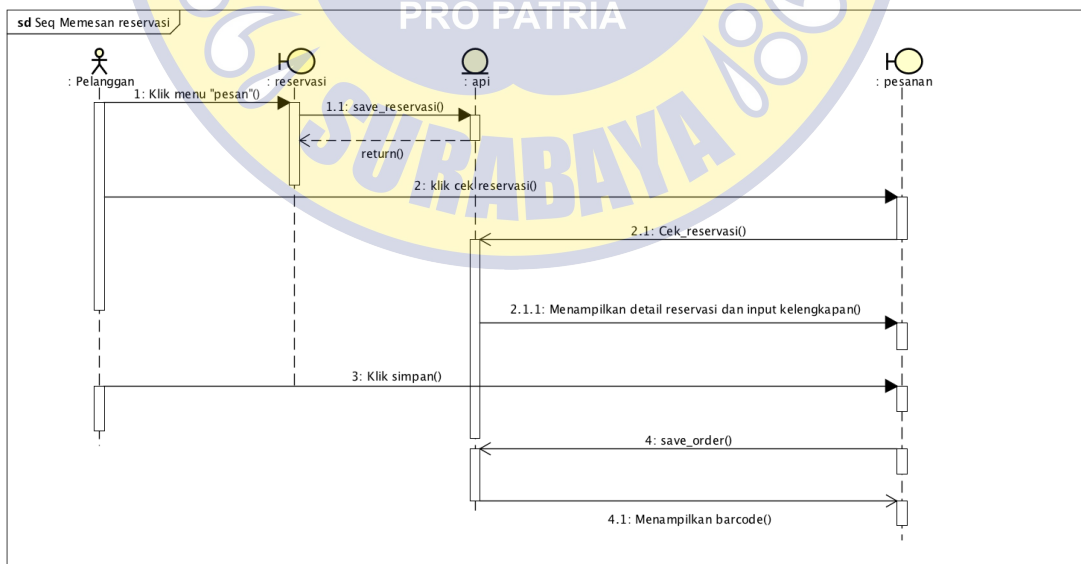
Gambar 4.33 Mengedit Item



Gambar 4.34 Menghapus Item

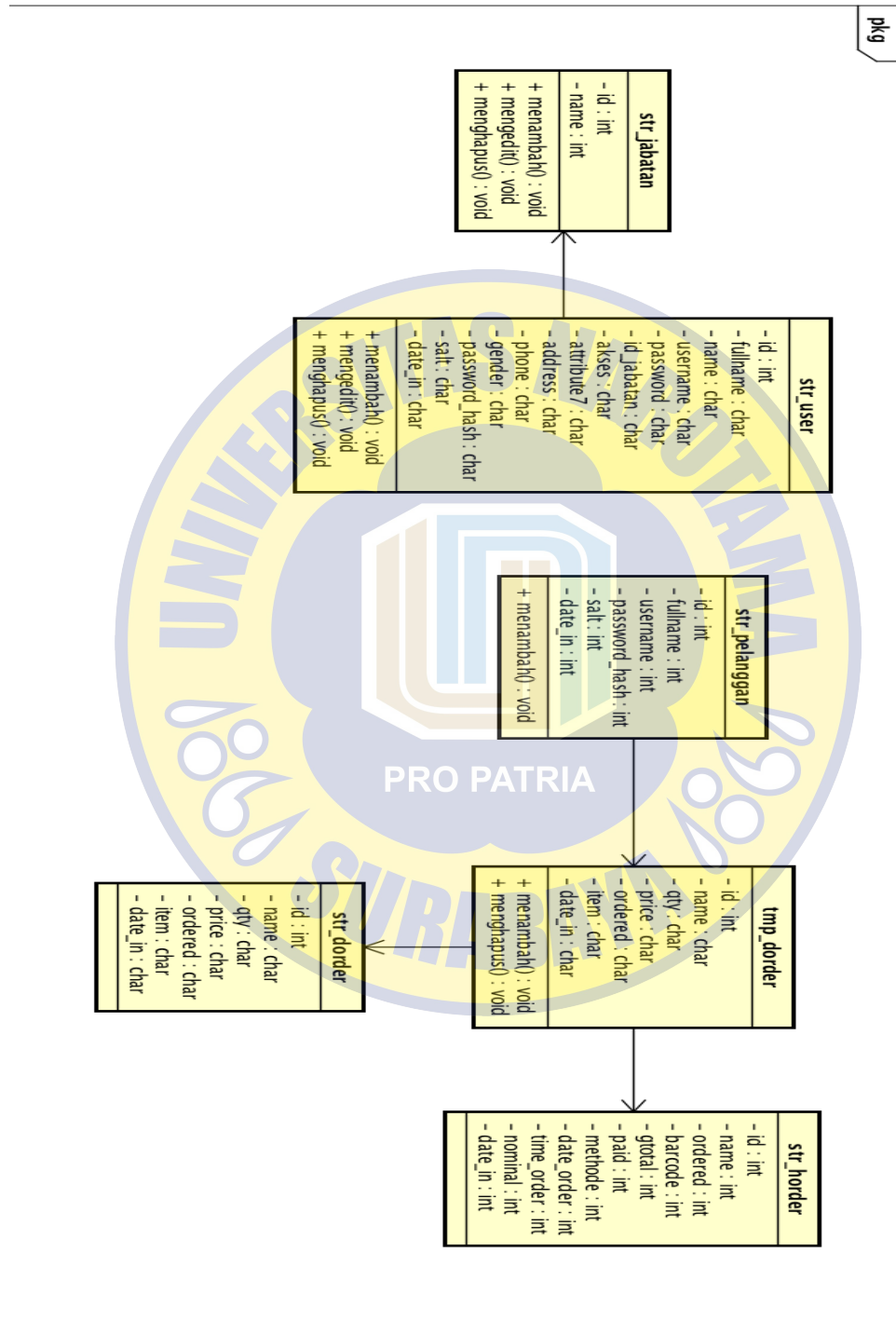


Gambar 4.35 Memesan Dine in



Gambar 4.36 Memesan Reservasi

4.2.7 Class Diagram



Gambar 4.37 Clas Diagram

4.3 User Interface



Gambar 4.38 UI Dine in dan Reservasi

Pada tampilan *mobile*, logo berada pada sisi kiri dan akan selalu tampil pada tiap menu yang dipilih. Pada gambar diatas adalah ketika pelanggan memilih menu *Dine In* atau reservasi, konten dibuat dua blok gambar dan keterangan minimalis untuk mempermudah pelanggan dalam melihat produk yang tersedia beserta nama produk dan harganya. Di setiap menu mempunyai konten produknya masing –

masing sehingga pelanggan dapat menentukan pemesanan khusus antara reservasi ataupun *Dine In*.



Gambar 4.39 Home

Pada gambar di atas merupakan tampilan *home*, untuk menampilkan semua produk yang ada, namun tidak untuk memesan. Tampilan pada menu *home*, dibuat tiga blok gambar minimalis dengan keterangan nama produk dan harga. Pemesanan

dikhususkan pada menu reservasi dan Dine In. Menu *home*, merupakan tampilan awal ketika pelanggan sudah melakukan login.

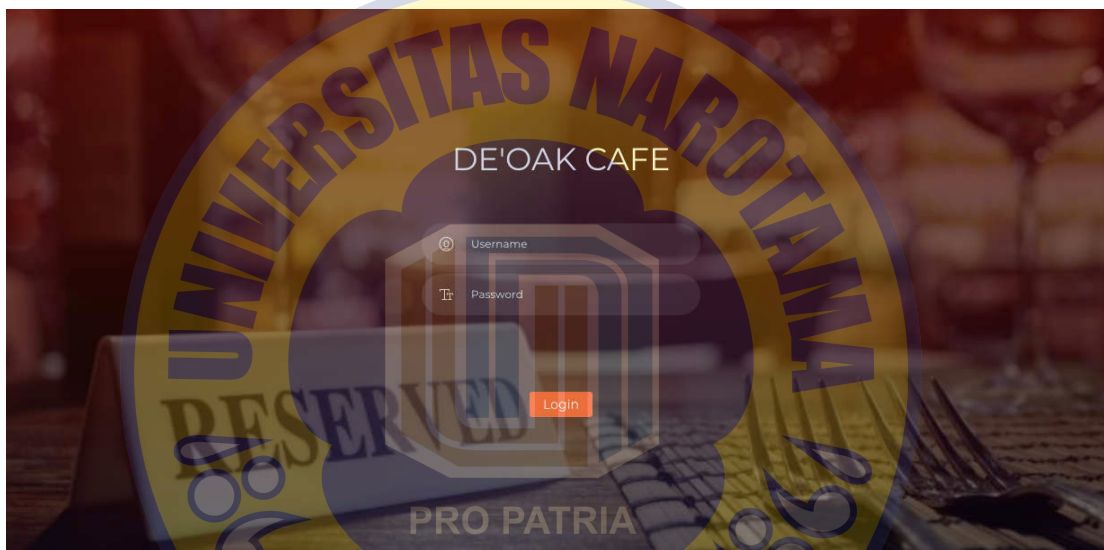


Gambar 4.40 Pesanan

Pada gambar diatas merupakan tampilan ketika pelanggan memilih menu pesanan. Pada menu ini, pelanggan dapat mengecek total pesanan dan mengambil *barcode* yang tersedia untuk melakukan transaksi kepada kasir.

4.4 Desain Program

Aplikasi yang digunakan dalam penelitian ini dibuat dengan menggunakan menggunakan bahasa pemrograman PHP dan JAVA. Peneliti menggunakan *framework* CodeIgniter yang menerapkan konsep MVC (*model, view, controller*) dan dalam proses penelitian, Penulis menggunakan *database* Mysql sebagai aplikasi *database* program. Berikut adalah contoh desain program yang digunakan dalam penelitian ini.



Gambar 4.41 halaman Login aplikasi

Gambar 4.41 menunjukkan halaman *login* aplikasi yang mana adalah tampilan utama pada aplikasi. Pada gambar tersebut berisi *username* dan *password user*.

De'Oak Cafe

kosong

Dashboard

User

Kategori

Item

Pesanan

Tambah Item

Nama

Harga

Foto

Choose File No file chosen

Pilih Kategori

Food

Detail Item

Tambah

Detail

Keterangan

Opsi

Simpan

Gambar 4.42 Halaman Modul Item

Gambar 4.42 menunjukkan menu dari aplikasi, isi dari aplikasi tersebut. Dalam menu tersebut, kasir dapat melakukan pengolahan *data* yang ada, yaitu mengelola item yang nantinya sebagai konten pada aplikasi android. Dalam tampilan menu aplikasi juga dapat menampilkan informasi-informasi untuk melakukan aktivitas pengolahan informasi item yang tersedia.

4.5 Implementasi

Pada bab ini menjelaskan tentang implementasi aplikasi dengan data nyata. Berikut adalah beberapa implementasi yang sudah dilakukan untuk mengisi konten pada aplikasi android dan desktop web.

De'Oak Cafe

kosong

Pesanan

Scan barcode

bibo210719103516day

Cari

Item	Qty	Harga
Roasty Tasty	1	Rp. 65000.00
Pizza Pepper Chicken	1	Rp. 65000.00

Total : Rp. 130000

Masukkan pembayaran :

Periksa :

Simpan

Gambar 4.43 Halaman konfirmasi pemesanan

Gambar 4.43 menunjukkan fungsi aplikasi pada saat konfirmasi pemesanan, yaitu dengan memberikan kode *barcode* ataupun dengan alat *scan barcode*. Konfirmasi pemesanan hanya bisa dilakukan ketika dilokasi.

Deoak Cafe

88% 14.25

PRO PATRIA

Username

Password

SIGN IN

Not Having an Account?

REGISTER

Gambar 4.44 Halaman login pada android 1

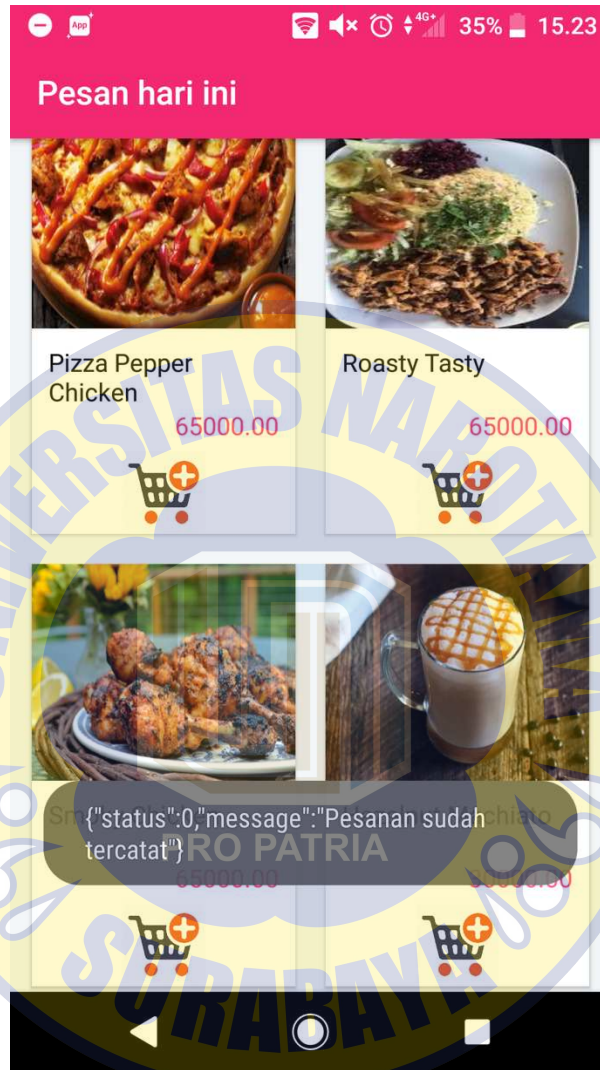
Gambar 4.44 merupakan halaman login pada aplikasi android. Pada tahap ini ketika pelanggan sudah melakukan login, aplikasi menyimpan data *preferences* dari *user*, sehingga ketika membuka aplikasi kembali tidak memerlukan login.



Gambar 4.45 Cek pesanan

Pada halaman ini berfungsi untuk pengecekan detail pesanan yang sudah dilakukan oleh pelanggan. Terdapat dua buah tombol yaitu “pesanan hari ini” dan pesanan reservasi. Pada kedua tombol tersebut mempunyai fungsi tersendiri yaitu pada tombol “pesanan hari ini” untuk mengecek pesanan untuk dine in saja,

sedangkan pesanan reservasi untuk mengecek pesanan reservasi saja.



Gambar 4.46 Cek pesanan

Pada gambar diatas dapat dijelaskan bahwa ketika pelanggan klik tombol pesan, maka akan keluar notifikasi pesanan sudah tercatat pada database yang tersedia pada server. Sehingga pelanggan dapat mengecek pesanan pada langkah berikutnya.

Langkah berikutnya yaitu, memilih menu pesanan, untuk mengecek pesanan yang sudah dipesan sebelumnya.



Gambar 4.47 cek pesanan dine in

Gambar diatas menjelaskan tentang pelanggan ketika cek pesanan dine in pada tombol pesanan hari ini, informasi pesanan akan keluar sesuai yang dipilih oleh pelanggan. Pada tahap selanjutnya pelanggan menyimpan pesanan untuk mendapatkan barcode sebagai validasi bahwa telah memesan.

Berikut adalah barcode yang didapat oleh pelanggan ketika telah menyimpan pesanan. Tahap selanjutnya adalah memberikan kode barcode tersebut kepada kasir untuk konfirmasi pesanan.



Gambar 4.48 Kode barcode

Tahap selanjutnya implementasi pada konfirmasi pesanan, kasir menerima kode barcode dari pelanggan.

Gambar 4.49 Konfirmasi pesanan

Kasir klik tombol cari pada kode barcode yang sudah terisi dan hasilnya akan keluar detail pesanan yang sudah dipesan sebelumnya.

4.6 Pengujian Aplikasi

Dalam tahap ini, penulis akan menguji aplikasi yang sudah dibuat. Pengujian aplikasi menggunakan *black box*, berikut adalah beberapa modul yang akan di uji :

Tabel 4.1 Modul pengujian

No	Nama Modul	Total modul
1.	Reservasi dine in	1 modul
2.	Konfirmasi pesanan	1 modul
3.	Item	3 modul

Peran modul yang diuji :

1. Ketika pelanggan akan melakukan reservasi aplikasi dapat menyimpan pesanan sesuai yang dipesan oleh pelanggan.
2. Pelanggan dapat mencetak barcode, ketika telah menyimpan pesanan
3. Ketika konfirmasi reservasi, kasir dapat menginputkan kode barcode untuk mengecek detail pesanan.
4. Modul item ketika menambah, edit dan hapus.

Dalam pengujian diharuskan melakukannya dengan cermat, sehingga pengujian aplikasi menjadi *valid* dan bila ditemukan kesalahan dalam aplikasi dapat diperbaiki dengan cepat.

Berikut adalah hasil pengujian black box

Tabel 4.2 pengujian blackbox 1

No	Modul pengujian	Defect yang didapat	Tipe kesalahan	Jenis Defect	Keterangan
1	Reservasi dine in	Tidak terdapat defect	Tidak ada kesalahan	Tidak terdapat defect	Modul berjalan dengan baik sesuai proses bisnis

2	Konfirmasi pesanan	Tidak terdapat defect	Tidak ada kesalahan	Tidak terdapat defect	Modul berjalan dengan baik, dengan data tersimpan pada database
3	Item	Tidak terdapat defect	Tidak ada kesalahan	Tidak terdapat defect	Modul berjalan dengan baik, dapat dirubah dan dihapus sesuai kinerja modul

4.7 Pembahasan

Penulis membangun aplikasi sesuai dengan kebutuhan yang ada pada DE'OAK CAFE. Pada tahapan ini penulis akan memaparkan hasil pembuatan desain *user interface* dan pembangunan aplikasi android dan desktop web base. Sesuai dengan metode desain yang digunakan yaitu UCD, dimana metode ini berfokus pada satu permasalahan yang kompleks pada sebuah kasus, sehingga permasalahan dapat diselesaikan dengan baik dan kebutuhan yang sesuai.

Pembangunan aplikasi desktop webbase menggunakan framework code igniter, dimana framework ini dapat dikembangkan lebih dalam lagi. Modul yang sudah dibuat pada aplikasi desktop web base berfungsi dengan baik, tidak ada bug yang tertinggal ataupun pengecekan struktur code yang salah. Pada pembuatan aplikasi android modul reservasi juga berfungsi dengan baik, sehingga pelanggan dapat memakai aplikasi tersebut dengan mudah tanpa harus mengantri depan kasir ataupun memanggil *waiters* untuk membuat pesanan. Berdasarkan pembuatan *user interface* dan pembangunan aplikasi, pengujian aplikasi berjalan dengan baik dengan metode Black BOX. Modul yang di uji merupakan fokus utama pada permasalahan yang ada pada DE'OAK CAFE yaitu terkait reservasi.

Reservasi yang sekarang sudah lebih optimal karena menggunakan aplikasi android sebagai pemesanan dan validasi menggunakan barcode, sehingga pemesanan bisa dilakukan kapanpun tanpa harus mengantri dan memanggil *waiters*.