

**PANDUAN LENGKAP APLIKASI JURAGAN
(JUALAN RAKYAT GABUNGAN *ONLINE*)
DENGAN *PLATFORM E-COMMERCE*
BERBASIS KOMUNITAS**



Dibuat Oleh,

1.16.4.089 Yusniar Nur Syarif Sidiq

**PROGRAM DIPLOMA IV TEKNIK INFORMATIKA
POLITEKNIK POS INDONESIA
BANDUNG
2019**

BAB V

ANALISIS DAN PERANCANGAN SISTEM

5.1 Analisis Dan Perancangan Sistem

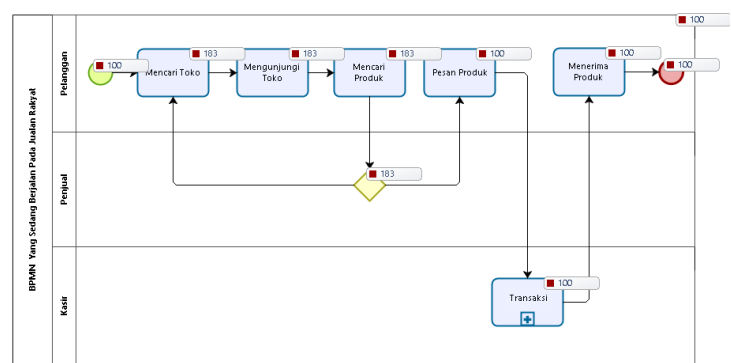
Analisis sistem adalah sebuah istilah yang secara kolektif mendeskripsikan fase-fase awal pengembangan sistem. Analisis sistem adalah teknik pemecahan masalah yang menguraikan bagian-bagian komponen dengan mempelajari seberapa bagus bagian-bagian komponen tersebut bekerja dan berinteraksi untuk mencapai tujuan mereka. Analisis sistem merupakan tahapan paling awal dari pengembangan sistem yang menjadi fondasi menentukan keberhasilan sistem informasi yang dihasilkan nantinya. Tahapan ini sangat penting karena menentukan bentuk sistem yang harus dibangun. Tahapan ini bisa merupakan tahapan yang mudah jika kalian tahu betul dari fungsionalitas dari sistem informasi yang akan dibuat.

5.1.1 Analisis Sistem Berjalan (*Current System*)

Hal yang dilakukan dalam pembuatan suatu sistem informasi adalah menganalisa sistem yang sedang di gunakan di lapangan. Dimana analisis sistem merupakan peroses menggali informasi dari suatu sistem, baik itu manual ataupun terkomputerisasi. Kemudian informasi tersebut diidentifikasi dan dievaluasi sehingga dapat

ditemukan permasalahan serta solusi untuk masalah tersebut yang kemudian dapat mengarah pada pembuatan sistem.

5.1.1.1 Analisis Flowmap Yang Sedang Berjalan



Gambar 5.1 BPMN Model Sistem Sedang Berjalan

Tabel 5.1 Hasil Simulasi Dari BPMN Sistem Sedang Berjalan

<i>Name</i>	<i>Type</i>	<i>Instances Completed</i>
BPMN Yang Sedang Berjalan Pada Jualan Rakyat	<i>Process</i>	<i>100</i>
<i>NoneStart</i>	<i>Start Event</i>	<i>100</i>
Mencari Toko	<i>Task</i>	<i>183</i>

Mengunjungi Toko	<i>Task</i>	<i>183</i>
Mencari Produk	<i>Task</i>	<i>183</i>
<i>ExvlisiveGateway</i>	<i>Gateway</i>	<i>183</i>
Pesan Produk	<i>Task</i>	<i>100</i>
Menerima Produk	<i>Task</i>	<i>100</i>
<i>NoneEnd</i>	<i>End event</i>	<i>100</i>
Transaksi	<i>Task</i>	<i>100</i>

5.1.1.2 Analisis Dokumen Yang Berjalan

Analisis dokumen yang digunakan merupakan tahap analisis terhadap dokumen yang terlibat yaitu dokumen toko – toko jualan rakyat. Dokumen tersebut terdiri dari data pedagang kaki lima, pengusaha mikro, dan pendiri UMKM yang berada pada Kota Bandung.

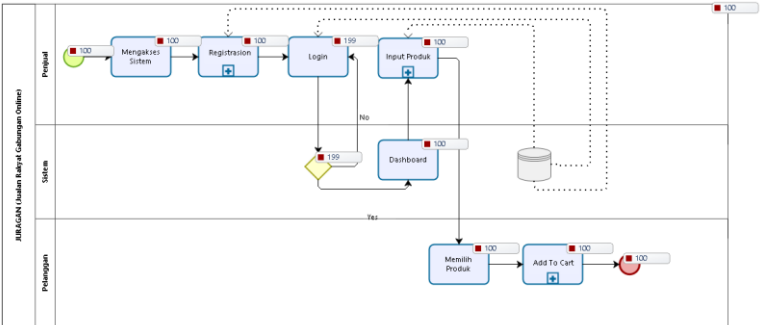
5.1.2 Analisis Sistem Yang Akan Dibangun

Dalam membangun sebuah sistem perlu diadakannya sebuah perancangan. Perancangan sistem ini dibuat dengan tujuan menghindari kesalahan dalam pembuatan sistem dan dapat berjalan semestinya sistem tersebut. Analisis sistem yang akan dibangun ini meliputi analisis prosedur

BPMN (*Business Process Model And Notation*) dan perancangan UML (*Unified Modelling Language*).

5.1.2.1 Analisis BPMN Yang Akan Dibangun

1) Analisis BPMN Pada Sistem Aplikasi JURAGAN



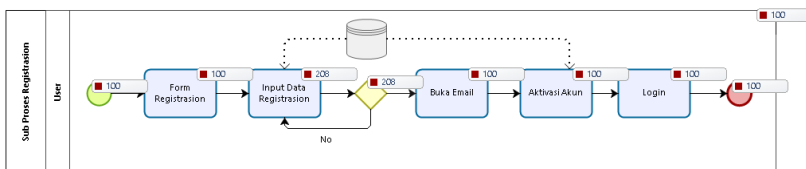
Gambar 5.2 BPMN Sistem JURAGAN

Tabel 5.2 Hasil Simulasi Dari BPMN Sistem JURAGAN

<i>Name</i>	<i>Type</i>	<i>Instances Completed</i>
Sistem JURAGAN	<i>Process</i>	<i>100</i>
<i>NoneStart</i>	<i>Start Event</i>	<i>100</i>
Mengakses Sistem	<i>Task</i>	<i>100</i>
<i>Login</i>	<i>Task</i>	<i>199</i>
<i>ExclusiveGateway</i>	<i>Gatewat</i>	<i>199</i>

<i>Dashboard</i>	<i>Task</i>	<i>100</i>
Memilih Produk	<i>Task</i>	<i>100</i>
<i>NoneEnd</i>	<i>End Event</i>	<i>100</i>
<i>Registration</i>	<i>Task</i>	<i>100</i>
Input Produk	<i>Task</i>	<i>100</i>
<i>Add To Cart</i>	<i>Task</i>	<i>100</i>

2) Analisis BPMN Pada Sub – Proses *Registration*



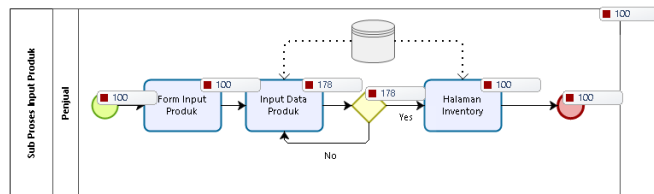
Gambar 5.3 BPMN Sub – Proses Registrasion

Tabel 5.3 Hasil Simulasi Dari BPMN Sub – Proses Registration

<i>Name</i>	<i>Type</i>	<i>Instances Completed</i>
Sub – Proses <i>Registration</i>	<i>Process</i>	<i>100</i>
<i>NoneStart</i>	<i>Start Event</i>	<i>100</i>

<i>Form Registrasion</i>	<i>Task</i>	<i>100</i>
<i>Input Data Registration</i>	<i>Task</i>	<i>208</i>
<i>ExclusiveGateway</i>	<i>Gateway</i>	<i>208</i>
<i>Buka Email</i>	<i>Task</i>	<i>100</i>
<i>Aktivasi Akun</i>	<i>Task</i>	<i>100</i>
<i>Login</i>	<i>Task</i>	<i>100</i>
<i>NoneEnd</i>	<i>End Event</i>	<i>100</i>

3) Analisis BPMN Sub – Proses Input Produk



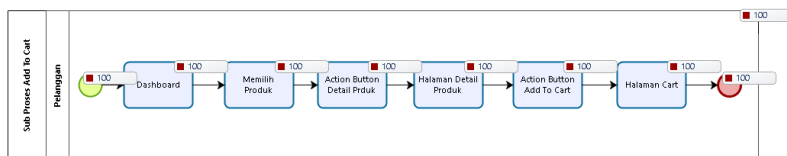
Gambar 5.4 BPMN Sub – Proses Input Produk

Tabel 5.4 Hasil Simulasi Dari BPMN Sub – Proses Inpu Produk

<i>Name</i>	<i>Type</i>	<i>Instances Completed</i>
-------------	-------------	----------------------------

Sub – Proses <i>Input</i> Produk	<i>Process</i>	100
<i>NoneStart</i>	<i>Start Event</i>	100
<i>Form Input</i> Produk	<i>Task</i>	100
<i>Input Data</i> Produk	<i>Task</i>	178
<i>ExclusiveGateway</i>	<i>Gateway</i>	178
Halaman <i>Inventory</i>	<i>Task</i>	100
<i>NoneEnd</i>	<i>End Event</i>	100

4) Analisis BPMN Sub – Proses *Add To Cart*



Gambar 5.5 BPMN Sub – Proses Add To Cart

Tabel 5.5 Hasil Simulasi Dari BPMN Sub – Proses Add To Cart

<i>Name</i>	<i>Type</i>	<i>Instances Completed</i>
-------------	-------------	----------------------------

Sub – Proses <i>Add To Cart</i>	<i>Process</i>	<i>100</i>
<i>NoneStart</i>	<i>Start Event</i>	<i>100</i>
<i>Dashboard</i>	<i>Task</i>	<i>100</i>
Memilih Produk	<i>Task</i>	<i>100</i>
<i>Action Button</i> Detail Produk	<i>Task</i>	<i>100</i>
Halaman Detail Produk	<i>Task</i>	<i>100</i>
<i>Action Button</i> <i>Add To Cart</i>	<i>Task</i>	<i>100</i>
Halaman <i>Cart</i>	<i>Task</i>	<i>100</i>
<i>NoneEnd</i>	<i>End Event</i>	<i>100</i>

5.1.2.2 Analisis Dokumen Yang Dibangun

Analisis dokumen tersebut merupakan tahap analisis dokumen yang akan terbentuk yaitu berupa dokumen data *user* yang terdiri dari data toko dan data produk. Adapun maksud dari dokumen tersebut seperti yang ditampilkan pada tabel 5.6 dan tabel 5.7.

Tabel 5.6 Dokumen Data Toko

Dibuat Oleh	Admin
Ditujukan Kepada	Admin
Isi Dokumen	Data toko yang terdaftar
Frekuensi	Dibuat sesuai data yang sudah terdaftar pada sistem
Tujuan Dokumen Dibuat	Mengolah data toko

Tabel 5.7 Dokumen Data Produk

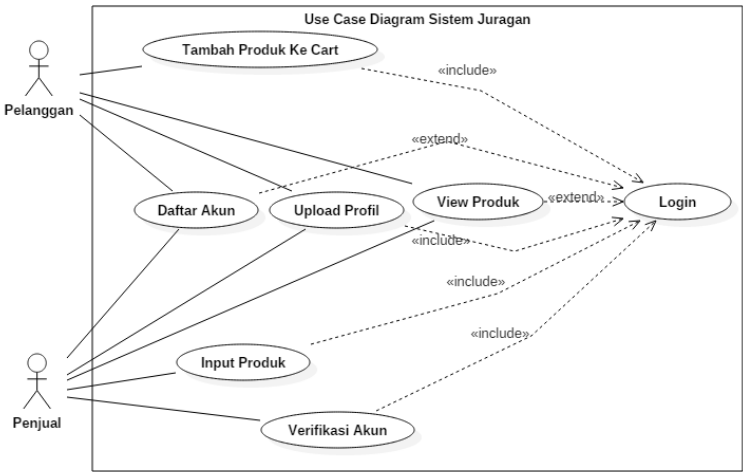
Dibuat Oleh	Penjual
Ditujukan Kepada	Pembeli
Isi Dokumen	Data produk yang telah di <i>input</i> kan
Frekuensi	Dibuat sesuai ketentuan sistem
Tujuan Dokumen Dibuat	Mempromosikan produk kepada pelanggan

5.1.2.3 UML (Unified Modelling Language)

5.1.2.3.1 Use Case Diagram

Use Case Diagram merupakan salah satu diagram yang diklasifikasikan ke dalam aspek

perilaku, dimana deskripsi perilaku dari setiap *use case* dijelaskan secara detail dan terpisah dengan menggunakan *document* secara *tekstual* yaitu *user case scanario*. Adapun *use case* dari sistem JURAGAN tersebut sebagai berikut :



Gambar 5.6 Use Case Diagram Sistem JURAGAN

5.1.2.3.1.1 Definisi Aktor

Tabel 5.8 Definisi Aktor

No.	Aktor	Deskripsi
1.	Pelanggan	a. Daftar Akun b. Login c. Upload Profil d. View Produk

		e. Tambah Produk Ke <i>Cart</i>
2.	Penjual	a. Daftar Akun b. <i>Login</i> c. <i>Upload</i> Profil d. <i>View</i> Produk e. <i>Input</i> Produk f. Verifikasi Akun

5.1.2.3.1.2 Definisi *Use Case*

Tabel 5.9 Definisi Use Case

No.	<i>Use Case</i>	Deskripsi
1.	Daftar Akun	a. Melakukan proses pendaftaran atau <i>registrasion</i> akun pada sistem JURAGAN.
2.	<i>Login</i>	a. Melakukan proses <i>login</i> terhadap akun yang sudah melakukan <i>registrasion</i> atau sudah terdaftar di

		dalam sistem JURAGAN.
3.	<i>View Produk</i>	<ul style="list-style-type: none"> a. Melakukan <i>view</i> terhadap produk yang sudah di <i>input</i> kan oleh penjual. b. Melihat detail produk secara keseluruhan.
4.	<i>Upload Profil</i>	<ul style="list-style-type: none"> a. Melakukan <i>input</i> dan <i>edit</i> profil terhadap akun pelanggan. b. Melakukan <i>input</i> dan <i>edit</i> profil terhadap akun toko.
5.	<i>Input Produk</i>	<ul style="list-style-type: none"> a. Melakukan proses <i>input</i> berupa informasi – informasi produk secara detail oleh penjual.
6.	Tambah Produk Ke <i>Cart</i>	<ul style="list-style-type: none"> a. Menambahkan produk yang ingin di beli ke dalam keranjang atau <i>cart</i>.

7	Verifikasi Akun	a. Melakukan verifikasi akun toko dengan menambahkan No. KTP dan Foto KTP.
---	-----------------	--

5.1.2.3.1.3 Use Case Scenario

Tabel 5.10 Use Case Scenario Daftar Akun

Identifikasi	
No.	JR1
Nama	Daftar Akun
Tujuan	Mendaftarkan akun pada sistem JURAGAN.
Deskripsi	Melakukan proses pendaftaran akun guna dapat melakukan proses <i>login</i> pada sistem JURAGAN.
Aktor	Pelanggan Dan Penjual
Skenario	
Kondisi Awal	<i>Display form registrasion</i>

Aksi Aktor	Reaksi Sistem
1. Akses <i>form registrasion</i>	a. <i>Display form registrasion</i>
2. Melakukan <i>input data registrasion</i> sesuai yang diinginkan oleh sistem	b. -
3. Memberikan <i>action</i> pada <i>button registrasion</i>	c. Memulai proses validasi
4. -	d. Mengirim <i>email</i> kepada <i>user</i> untuk melakukan proses verifikasi
5. Mendapatkan <i>email</i>	e. -
6. Melakukan proses verifikasi akun	f. Proses verifikasi
7. -	g. Mengaktifkan akun

8. Menerima notif akun aktif	h. -
---------------------------------	------

Tabel 5.11 Use Case Scenario Login

Identifikasi	
No.	JR2
Nama	<i>Login</i>
Tujuan	Mengakses sistem JURAGAN.
Deskripsi	Mengakses sistem JURAGAN secara menyeluruh dengan kategori <i>user</i> tertentu.
Aktor	Pelanggan Dan Penjual
Skenario	
Kondisi Awal	<i>Display form login</i>
Aksi Aktor	Reaksi Sistem
1. Melakukan <i>input emali</i> dan <i>password</i> yang sudah terdaftar pada sistem	a. -

2. Memberikan <i>action</i> pada <i>button login</i>	b. Memulai proses validasi
3. -	c. Proses verifikasi
4. -	d. <i>Display</i> <i>dashboard</i> sistem JURAGAN

*Tabel 5.12 Use Case Scenario View Produk
Pelanggan*

Identifikasi	
No.	JR3
Nama	<i>View Produk Pelanggan</i>
Tujuan	Melihat keterangan produk.
Deskripsi	Melihat informasi – informasi produk yang telah di <i>upload</i> oleh penjual secara lebih detail.
Aktor	Pelanggan

Skenario	
Kondisi Awal	<i>Dashboard</i>
Aksi Aktor	Reaksi Sistem
1. Memberikan <i>action</i> pada <i>button</i> detail pada produk yang pelanggan inginkan	a. <i>Display</i> detail produk

Tabel 5.13 Use Case Scenario View Produk Penjual

Identifikasi	
No.	JR4
Nama	<i>View Produk Penjual</i>
Tujuan	Melihat keterangan produk.
Deskripsi	Melihat detail produk
Aktor	Penjual
Skenario	
Kondisi Awal	<i>Display Inventory</i>
Aksi Aktor	Reaksi Sistem

1. Memberikan <i>action</i> pada <i>button</i> detail	a. <i>Display</i> detail produk
---	------------------------------------

*Tabel 5.14 Use Case Scenario Edit Profil
Pelanggan*

Identifikasi	
No.	JR5
Nama	<i>Edit</i> Profil Pelanggan
Tujuan	Melakukan proses edit
Deskripsi	Melakukan edit terhadap profil pelanggan
Aktor	Pelanggan
Skenario	
Kondisi Awal	<i>Display edit profile</i>
Aksi Aktor	Reaksi Sistem
1. Merubah atau melakukan <i>edit</i> pada data profil	a. -
2. Memberikan <i>action</i> pada <i>button</i> <i>edit</i> <i>profile</i>	b. Memulai proses validasi

3. -	c. Proses verifikasi
4. -	d. Melakukan proses <i>upload</i> ke <i>database</i>
5. Menerima data baru	e. Menampilkan data baru

Tabel 5.15 Use Case Scenario Edit Profil Penjual

Identifikasi	
No.	JR6
Nama	<i>Edit Profil Penjual</i>
Tujuan	Melakukan proses edit
Deskripsi	Melakukan edit terhadap profil penjual
Aktor	Penjual
Skenario	
Kondisi Awal	<i>Display My Store</i>
Aksi Aktor	Reaksi Sistem
1. Merubah atau melakukan <i>edit</i> pada data profil	a. -

2. Memberikan <i>action</i> pada <i>button edit</i> <i>profile</i>	b. Memulai proses validasi
3. -	c. Proses verifikasi
4. -	d. Melakukan proses <i>upload</i> ke <i>database</i>
5. Menerima data baru	e. Menampilkan data baru

Tabel 5.16 Use Case Scenario Changed Password

Identifikasi	
No.	JR7
Nama	<i>Changed Password</i>
Tujuan	Merubah password
Deskripsi	Mengganti password lama menjadi password baru
Aktor	Pelanggan dan Penjual
Skenario	

Kondisi Awal	<i>Display Change Password</i>
Aksi Aktor	Reaksi Sistem
1. <i>Input password lama</i>	a. -
2. <i>Input password baru</i>	b. -
3. Memberikan <i>action</i> pada <i>button change password</i>	c. Memulai proses validasi
4. -	d. Proses verifikasi
5. -	e. <i>Database</i> melakukan fungsi <i>update</i> sehingga <i>password</i> berganti
6. Mendapatkan notif berhasil ganti password	f. Memberikan notif

Tabel 5.17 Use Case Scenario Input Produk

Identifikasi

No.	JR8
Nama	<i>Input Produk</i>
Tujuan	Mempromosikan produk
Deskripsi	Melakukan <i>input</i> produk agar produk dapat tampil di halaman <i>dashboard</i>
Aktor	Penjual
Skenario	
Kondisi Awal	<i>Display Inventory</i>
Aksi Aktor	Reaksi Sistem
1. Memberikan <i>action</i> pada <i>button add items</i>	a. Memunculkan <i>form input</i> produk dengan fungsi modal pada <i>bootstrap</i>
2. Melakukan <i>input</i> data produk	b. -
3. Memberikan <i>action</i> pada <i>button save</i> <i>product</i>	c. Memulai proses validasi
4. -	d. Proses verifikasi

5. -	e. <i>Database</i> melakukan proses <i>upload</i>
6. Menerima data produk	f. Menampilkan data produk

Tabel 5.18 Use Case Scenario Tambah Produk Ke

Cart

Identifikasi	
No.	JR9
Nama	Tambah Produk Ke <i>Cart</i>
Tujuan	Menambahkan produk ke dalam <i>cart</i>
Deskripsi	Produk yang telah di <i>input</i> kan oleh penjual akan di tambahkan kedalam <i>cart</i>
Aktor	Pelanggan
Skenario	
Kondisi Awal	<i>Dashboard</i>
Aksi Aktor	Reaksi Sistem
1. Memilih produk	a.

2. Melihat informasi produk	b. <i>Display</i> detail produk
3. Memberikan <i>action</i> pada <i>button add to cart</i>	c. Menambahkan produk ke dalam <i>cart</i>

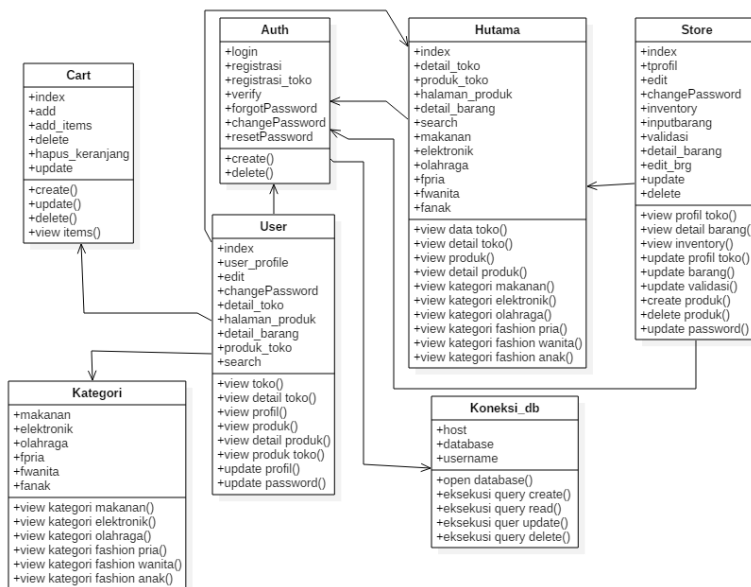
Tabel 5.19 Use Case Scenario Verifikasi Akun

Identifikasi	
No.	JR10
Nama	Verifikasi Akun
Tujuan	Mengaktifkan akun toko
Deskripsi	Akun toko akan muncul dihalaman pelanggan apabila sudah terverifikasi dengan memasukkan No. KTP dan Foto KTP.
Aktor	Penjual
Skenario	
Kondisi Awal	Halaman Verifikasi
Aksi Aktor	Reaksi Sistem

1. <i>Input</i> No. KTP	a. -
2. <i>Upload</i> foto KTP	b. -
3. Tekan <i>button</i> verifikasi akun	c. <i>Validasi</i>
4. -	d. <i>Update</i> status toko

5.1.2.3.2 ***Class Diagram***

Class diagram adalah sebuah spesifikasi yang jika tidak diinstansiasi akan menghasilkan sebuah obyek dan merupakan inti dari pengembangan dan desain berorientasi obyek. Adapun *class* diagram dari sistem JURAGAN adalah sebagai berikut :

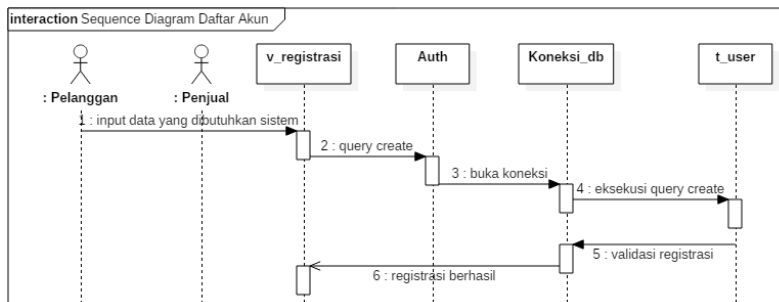


Gambar 5.7 Class Diagram Sistem JURAGAN

5.1.2.3.3 Sequence Diagram

Sequence diagram adalah suatu diagram yang memperlihatkan atau menampilkan interaksi – interaksi antar objek di dalam sistem yang disusun pada sebuah urutan atau rangkaian waktu. Adapun *sequence* diagram dari sistem JURAGAN adalah sebagai berikut:

1) *Sequence Diagram Daftar Akun*

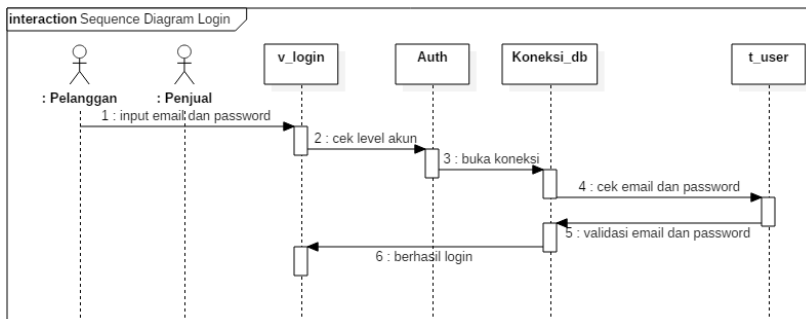


Gambar 5.8 Sequence Diagram Daftar Akun

Keterangan :

1. Pelanggan dan Penjual akan melakukan *input* data yang dibutuhkan.
2. *View registrasi* akan membuat *query create* melalui *controller Auth*.
3. *Controller Auth* akan membuka koneksi ke *database*.
4. Eksekusi *query create* akan di lakukan kepada tabel *user*.
5. Validasi *registrasi* akan di lakukan oleh tabel *user*.
6. *Registrasi* berhasil

2) *Sequence Diagram Login*

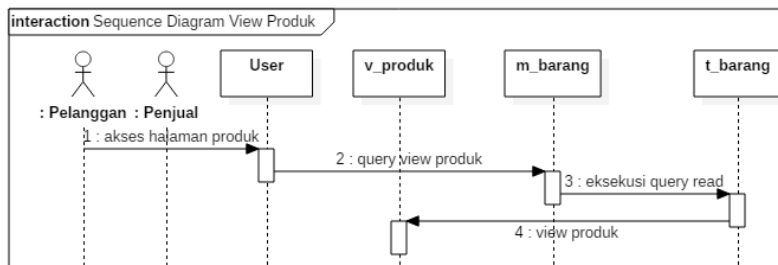


Gambar 5. 9 Sequence Diagram Login

Keterangan :

1. Pelanggan dan Penjual akan melakukan *input email dan password*.
2. Sistem akan melakukan cek level akun pada *controller Auth*.
3. *Controller Auth* akan membuka koneksi *database*.
4. Di dalam *database* akan dilakukan proses pengecekan *email* atau *password* yang sudah terdaftar.
5. *Database* juga akan melakukan *validasi* terhadap tabel *user*.
6. Apabila lolos maka *user* berhasil melakukan *login*.

3) Sequence Diagram View Produk

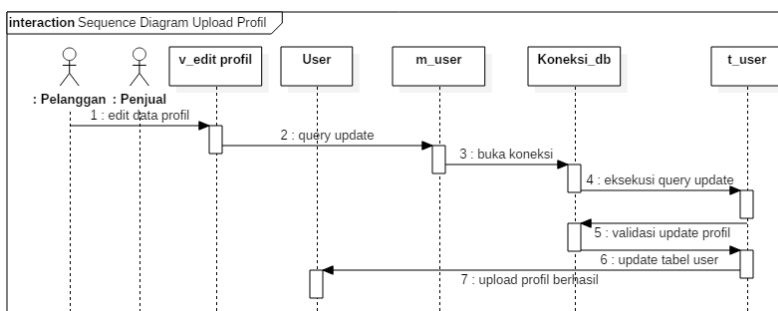


Gambar 5.10 Sequence Diagram View Produk Pelanggan

Keterangan :

1. *User* akan melakukan akses terhadap halaman produk.
2. *Controller* auth akan menampilkan halaman produk dan mengirimkan *query view* produk.
3. Pada model barang dimana *query view* tersebut di eksekusi kepada tabel barang.
4. Tabel barang akan melakukan *view* produk ke halaman *view* produk.

4) *Sequence Diagram Upload Profil*

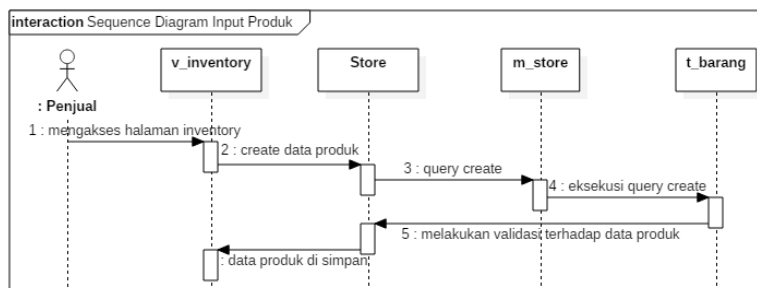


Gambar 5.11 Sequence Diagram Upload Profil

Keterangan :

1. *User* akan mengakses halaman *edit* profil dan merubah profil mereka pada halaman tersebut.
2. *User* akan menekan *button edit* dan *query update* akan dikirim kepada model *user*.
3. *Model user* akan melakukan akses terhadap *database*.
4. *Database* akan melakukan eksekusi *query update* kepada tabel *user*.
5. Tabel *user* akan melakukan validasi.
6. *Database* akan merubah data pada tabel *user* melalui *query update*.
7. Data akan tersimpan pada tabel *user* dan *update* profil berhasil.

5) Sequence Diagram Input Produk

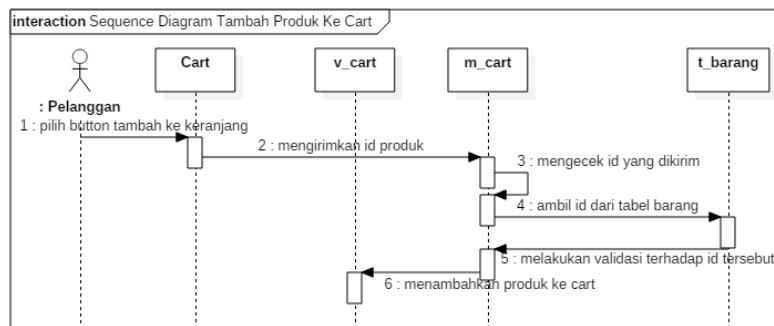


Gambar 5.12 Sequence Diagram Input Produk

Keterangan :

1. Penjual akan melakukan akses terhadap halaman *inverntory*.
2. Pada halaman *inventory* tersebut pelanggan akan melakan *create* produk.
3. *Controller* akan melakukan mengirimkan *query create* kepada model *store*.
4. *Query create* tersebut akan dieksekusi oleh model *store* kepada *tabel barang*.
5. *Controller Store* akan melakukan validasi terhadap data produk yang dikirim.
6. Produk akan tersimpan di tabel barang dan ditampilkan pada halaman *inventory*.

6) Sequence Diagram Tambah Produk Ke Cart

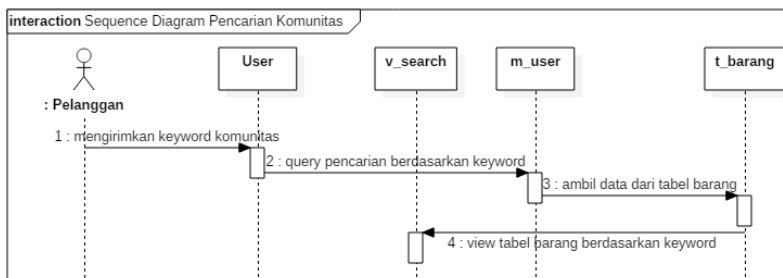


Gambar 5.13 Sequence Diagram Tambah Produk Ke Cart

Keterangan :

1. Pelanggan akan menekan *button* tambah ke keranjang pada sebuah produk yang *id* dari produk tersebut akan dikirim.
2. *Controller cart* akan mengirimkan *id* ke dalam *model cart*.
3. *Model cart* akan mengecek *id* yang dikirim.
4. *Model cart* akan mengambil *id* tersebut dari tabel barang
5. Validasi pada *id* tersebut akan dilakukan.
6. Apabila lolos maka *model cart* akan menambahkan produk tersebut ke halaman *cart*.

7) Sequence Diagram Pencarian Komunitas



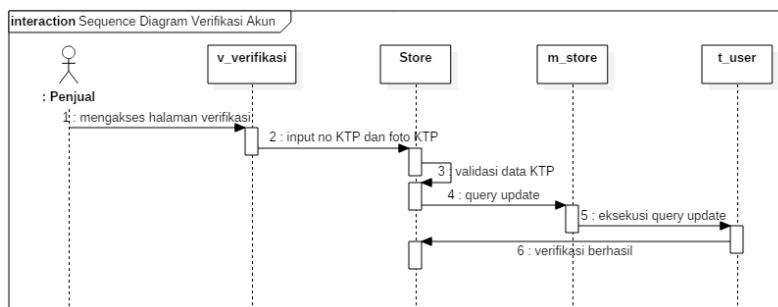
Gambar 5.14 Sequence Diagram Pencarian Komunitas

Keterangan :

1. Pelanggan akan mengirimkan *keyword* komunitasnya.

2. Pada *controller user* dimana akan mengirimkan *query view* dengan menambahkan *keyword* yang di cari.
3. *Model user* akan mengambil data dari tabel barang.
4. Tabel barang akan melakukan *view* sesuai *keyword* yang diberikan.

8) Sequence Diagram Verifikasi Akun



Gambar 5.15 Sequence Diagram Verifikasi Akun

Keterangan :

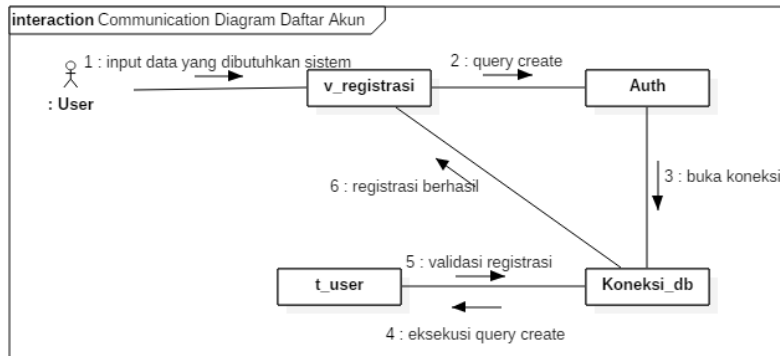
1. Penjual akan mengakses halaman verifikasi.
2. Pada halaman verifikasi dimana penjual perlu mengisi No KTP dan melakukan *upload* foto KTP.
3. Pada *controller store* dimana akan terjadinya proses validasi data KTP.

4. *Controller Store* akan mengirimkan *query update* kepada *model store*.
5. *Model store* akan melakukan eksekusi *query update* terhadap tabel *user*.
6. Verifikasi akun toko telah berhasil dilakukan.

5.1.2.3.4 Collaboration Diagram

Collaboration diagram yaitu dimana sebuah pengelompokkan pesan terhadap kumpulan diagram sekuen menjadi sebuah diagram. Pada *collaboration* diagram dimana terdapat *method* yang dijalankan antara objek yang satu dan objek lainnya. Dalam *collaboration* diagram tersebut diaman objek harus melakukan sinkronasi pesan dengan serangkaian pesan – pesan lainnya. Adapun *Collaboration* diagram pada sistem JURAGAN adalah sebagai berikut :

1) Collaboration Diagram Daftar Akun

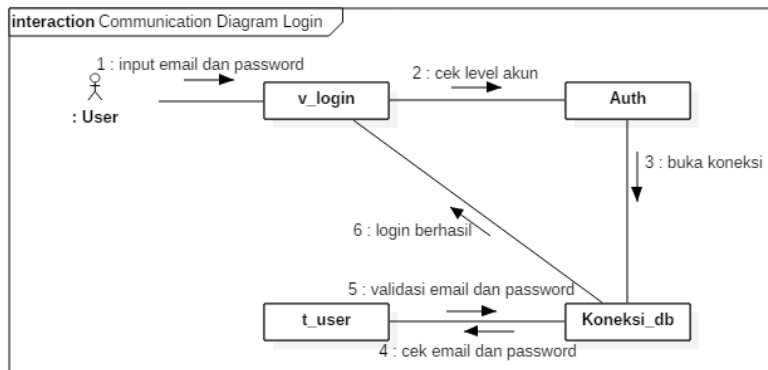


Gambar 5.16 Collaboration Diagram Daftar Akun

Keterangan :

1. Pelanggan dan Penjual akan melakukan *input* data yang dibutuhkan.
2. *View registrasi* akan membuat *query create* melalui *controller Auth*.
3. *Controller Auth* akan membuka koneksi ke *database*.
4. Eksekusi *query create* akan di lakukan kepada tabel *user*.
5. Validasi *registrasi* akan di lakukan oleh tabel *user*.
6. *Registrasi* berhasil.

2) Collaboration Diagram Login

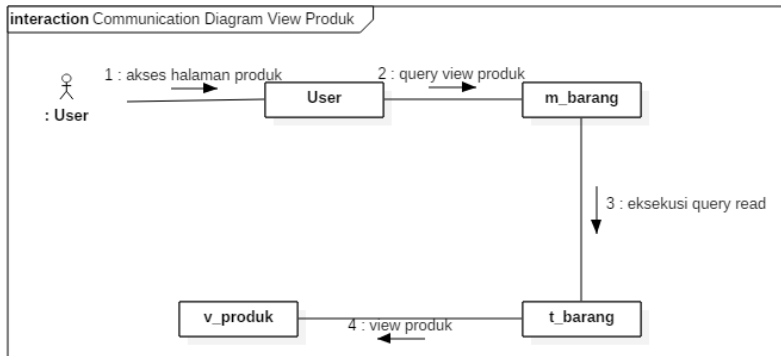


Gambar 5.17 Collaboration Diagram Login

Keterangan :

1. Pelanggan dan Penjual akan melakukan *input email* dan *password*.
2. Sistem akan melakukan cek level akun pada *controller Auth*.
3. *Controller Auth* akan membuka koneksi *database*.
4. Di dalam *database* akan dilakukan proses pengecekan *email* atau *password* yang sudah terdaftar.
5. *Database* juga akan melakukan *validasi* terhadap tabel *user*.
6. Apabila lolos maka *user* berhasil melakukan *login*.

3) Collaboration Diagram View Produk

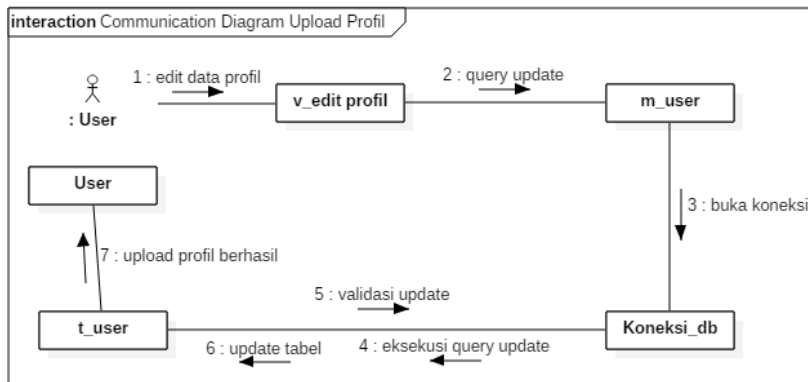


Gambar 5.18 Collaboration Diagram View Produk

Keterangan :

1. *User* akan melakukan akses terhadap halaman produk.
2. *Controller* auth akan menampilkan halaman produk dan mengirimkan *query view* produk.
3. Pada model barang dimana *query view* tersebut di eksekusi kepada tabel barang.
4. Tabel barang akan melakukan *view* produk ke halaman *view* produk

4) Collaboration Diagram Upload Profil

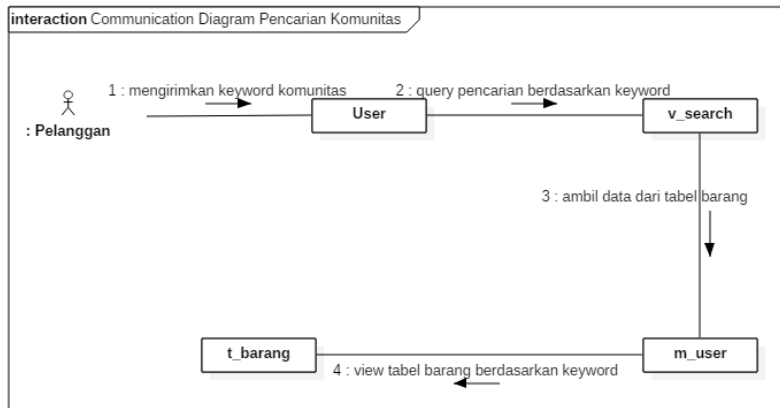


Gambar 5.19 Collaboration Diagram Upload Profil

Keterangan :

1. *User* akan mengakses halaman *edit* profil dan merubah profil mereka pada halaman tersebut.
2. *User* akan menekan *button edit* dan *query update* akan dikirim kepada model *user*.
3. *Model user* akan melakukan akses terhadap *database*.
4. *Database* akan melakukan eksekusi *query update* kepada tabel *user*.
5. Tabel *user* akan melakukan validasi.
6. *Database* akan merubah data pada tabel *user* melalui *query update*.
7. Data akan tersimpan pada tabel *user* dan *update* profil berhasil.

5) Collaboration Diagram Pencarian Komunitas

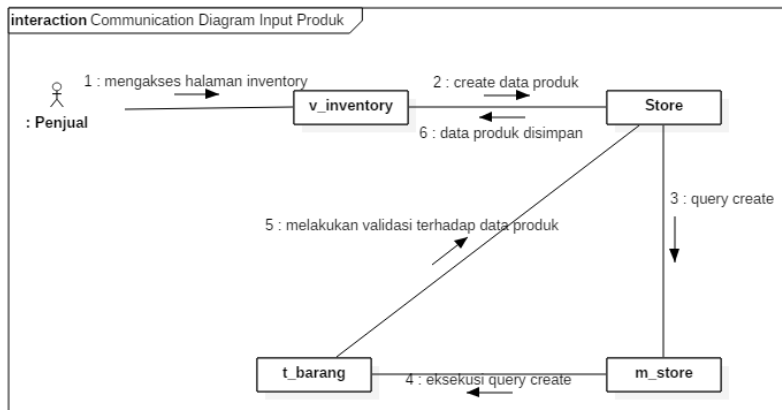


Gambar 5.20 Collaboration Diagram Pencarian Komunitas

Keterangan :

1. Pelanggan akan mengirimkan *keyword* komunitasnya.
2. Pada *controller user* dimana akan mengirimkan *query view* dengan menambahkan *keyword* yang di cari.
3. *Model user* akan mengambil data dari tabel barang.
4. Tabel barang akan melakukan *view* sesuai *keyword* yang diberikan.

6) Collaboration Diagram Input Produk

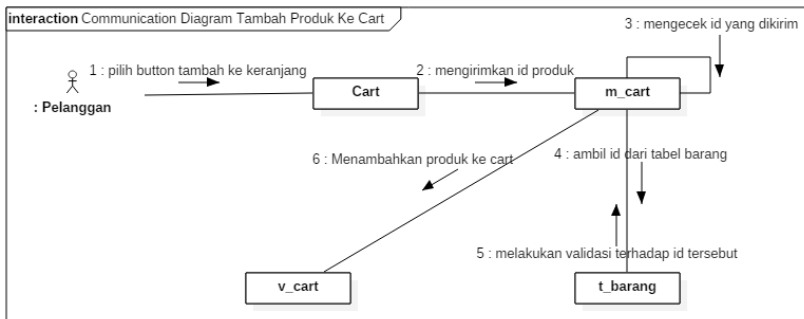


Gambar 5.21 Collaboration Diagram Input Produk

Keterangan :

1. Penjual akan melakukan akses terhadap halaman *inverntory*.
2. Pada halaman *inventory* tersebut pelanggan akan melakan *create* produk.
3. *Controller* akan melakukan mengirimkan *query create* kepada model *store*.
4. *Query create* tersebut akan dieksekusi oleh model *store* kepada *tabel barang*.
5. *Controller Store* akan melakukan validasi terhadap data produk yang dikirim.
6. Produk akan tersimpan di *tabel barang* dan ditampilkan pada halaman *inventory*.

7) Collaboration Diagram Tambah Produk Ke Cart

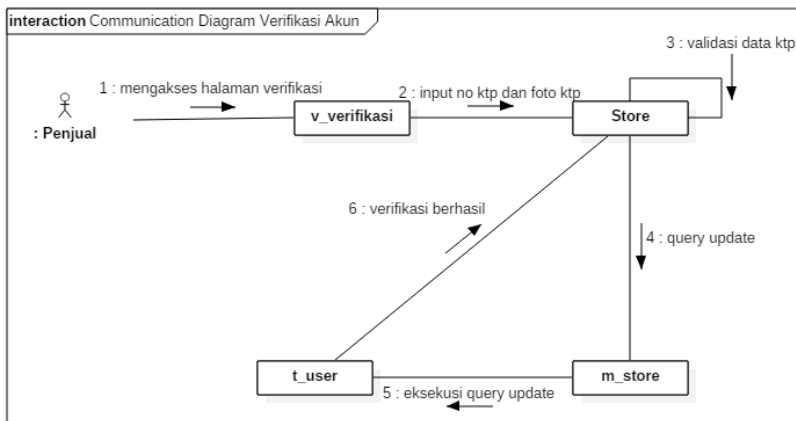


Gambar 5.22 Collaboration Diagram Tambah Produk Ke Cart

Keterangan :

1. Pelanggan akan menekan *button* tambah ke keranjang pada sebuah produk yang *id* dari produk tersebut akan dikirim.
2. *Controller cart* akan mengirimkan *id* ke dalam *model cart*.
3. *Model cart* akan mengecek *id* yang dikirim.
4. *Model cart* akan mengambil *id* tersebut dari tabel barang
5. Validasi pada *id* tersebut akan dilakukan.
6. Apabila lolos maka *model cart* akan menambahkan produk tersebut ke halaman *cart*.

8) Collaboration Diagram Verifikasi Akun



Gambar 5.23 Collaboration Diagram Verifikasi Akun

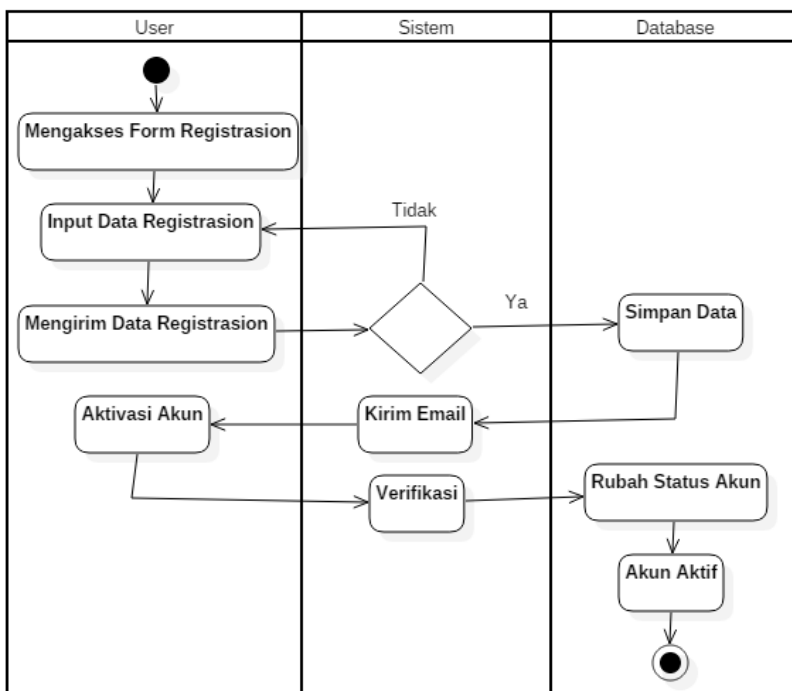
Keterangan :

1. Penjual akan mengakses halaman verifikasi.
2. Pada halaman verifikasi dimana penjual perlu mengisikan No KTP dan melakukan *upload* foto KTP.
3. Pada *controller store* dimana akan terjadinya proses validasi data KTP.
4. *Controller Store* akan mengirimkan *query update* kepada *model store*.
5. *Model store* akan melakukan eksekusi *query update* terhadap tabel *user*.
6. Verifikasi akun toko telah berhasil dilakukan.

5.1.2.3.5 Activity Diagram

Secara umum *activity* diagram merupakan gambaran alur dari suatu sistem yang dibuat, sehingga pengguna mengerti kegunaan sistem yang akan dibangun. *Activity* diagram merupakan gambaran *workflow* atau aktivitas dari sebuah sistem dalam proses bisnis.

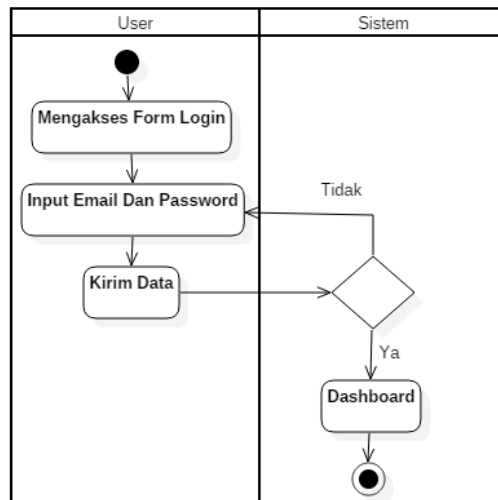
1) Activity Diagram Registrasion



Gambar 5.24 Activity Diagram Registrasion

Pada *activity* diagram tersebut dimana dijelaskan bahwa *user* akan mengakses terlebih dahulu *form registrasion*. Setelah *form registrasion* terbuka maka langkah selanjutnya dimana *user* akan melakukan *input* data *registrasion* pada *form* tersebut sesuai dengan ketentuan yang diberikan oleh sistem. Data yang di *nput* kan tersebut akan dikirim kepada sistem oleh *user*. Sistem akan melakukan proses validasi dan apabila berhasil maka data akan di simpan kedalam *database* namun akun belum aktif sehingga *user* tidak dapat melakukan *login*. Untuk mengaktifkan akun tersebut dimana *user* perlu melakukan aktivasi akun pada *email* yang telah dikirimkan oleh sistem. Sistem akan melakukan verifikasi. Data yang berada pada *database* akan di *update* sehingga status akun berubah menjadi aktif.

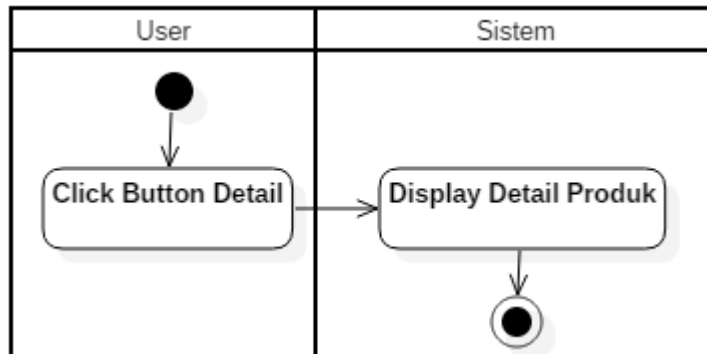
2) Activity Diagram Login



Gambar 5.25 Activity Diagram Login

Pada *activity* diagram tersebut dimana *step* pertama yang *user* lakukan adalah mengakses *form login*. Ketika *form login* sudah terakses maka *step* berikutnya *user* akan melakukan *input email* dan *password* yang sudah terdaftar pada sistem. Data *email* dan *password* yang sudah di *input* kan tersebut akan dikirimkan kepada sistem. Dimana sistem akan melakukan proses validasi dan apabila berhasil maka sistem akan mengarahkan *user* ke dalam *dashboard* yang menandakan bahwa *login* telah berhasil di lakukan.

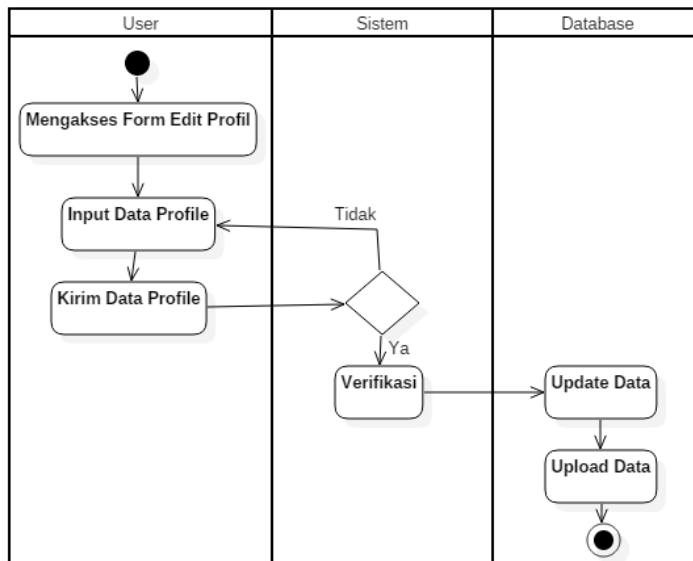
3) Activity Diagram View Produk



Gambar 5.26 Activity Diagram View Produk

Pada *activity* diagram tersebut dimana menunjukkan aktivitas *user* dalam melakukan *view* produk. Hal pertama yang harus *user* lakukan adalah memberikan *action* terhadap *button* detail dengan cara meng *click button* tersebut. Ketika *button* tersebut di *click* dimana sistem akan mengirim *id* dari produk tersebut dan memunculkan *display* detail produk. Dalam *display* detail produk dimana akan menampilkan informasi – informasi secara detail pada produk tersebut.

4) Activity Diagram *Edit Profile*

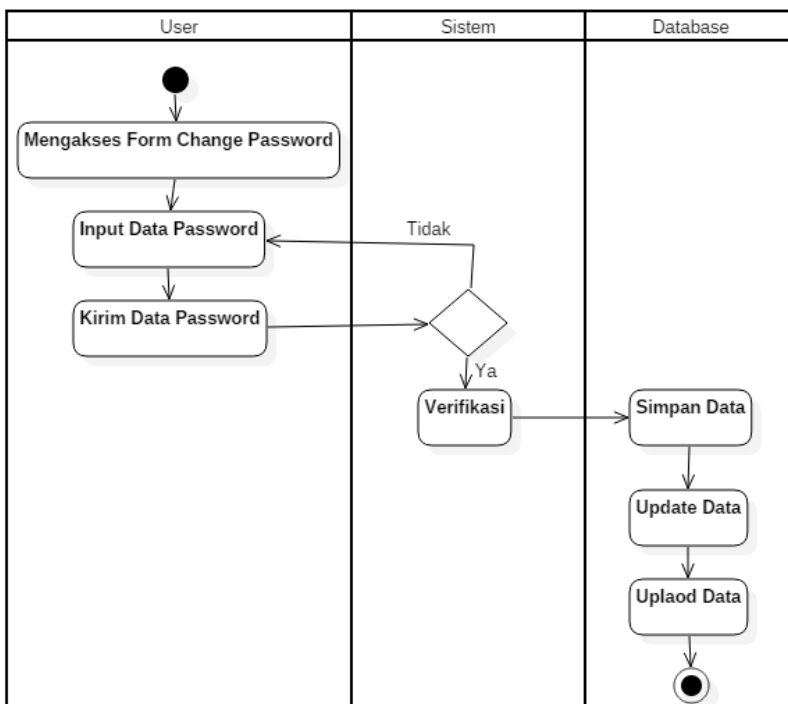


Gambar 5.27 Activity Diagram Edit Profile

Pada *activity* diagram tersebut dimana menunjukkan sebuah aktifitas *edit profile*. Dimana *step* pertama dalam melakukan aktifitas *edit profile* *user* perlu mengakses *form edit profile*. Pada *form edit profile* tersebut *user* akan melakukan perubahan data *profile*-nya. Ketika data sudah dirubah maka *user* akan mengirimkan data tersebut kepada sistem. Sistem akan melakukan proses validasi dimana untuk mengecek data tersebut apakah sudah benar. Jika data sudah benar maka sistem akan melakukan verifikasi. Setelah data terverifikasi sistem akan mengirimkan

data tersebut kepada *database*. Di dalam *database* dimana terjadi fungsi *get* yang dimana untuk melakukan *update* data. Data yang sudah di *update* akan di *upload* agra *user* dapat melihat perubahannya.

5) Activity Diagram Change Password

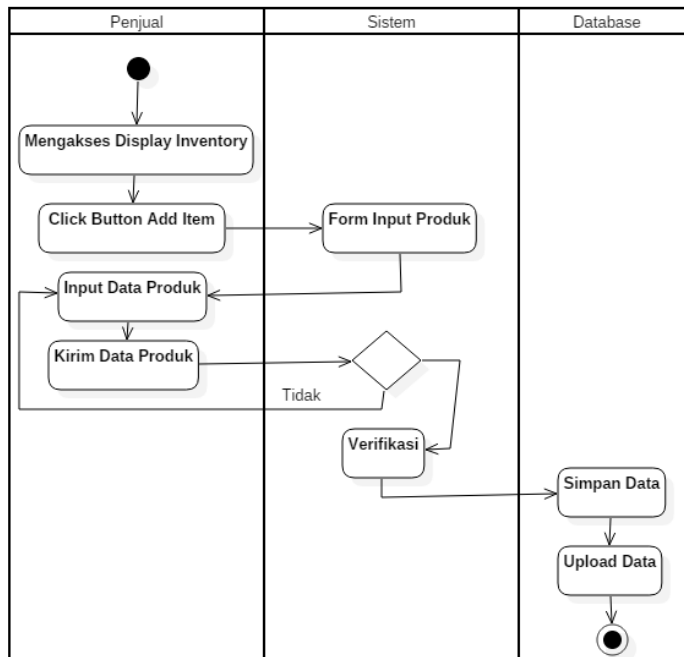


Gambar 5.28 Activity Diagram Change Password

Pada *activity* diagram tersebut dimana menjelaskan mengenai aktifitas dalam melakukan *change password*. Step pertama yang perlu *user* lakukan adalah mengakses *form change password*.

Pada *form* tersebut dimana dibutuhkan data berupa *password* lama dan *password* baru yang dimana *user* perlu melakukan *input* data tersebut. Data yang di *input* kan tersebut akan dikirimkan kepada sistem oleh *user*. Sistem akan melakukan validasi untuk mengecek data apakah sudah benar. Apabila sudah benar sistem akan memverifikasi data tersebut. Data yang sudah terverifikasi akan dikirimkan ke dalam *database*. Pada *database* akan dilakukan sebuah fungsi *get* yang berguna untuk melakukan *update* data. Data yang sudah di *update* akan di *upload* oleh *database*.

6) Activity Diagram *Input Produk*

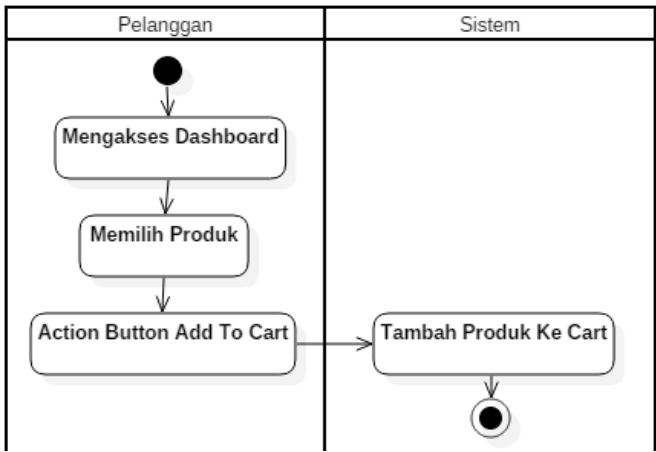


Gambar 5.29 Activity Diagram *Input Produk*

Pada *activity diagram* tersebut dimana menjelaskan bagaimana aktifitas pada proses *input data produk* dan aktor yang berperan adalah *user* dengan level penjual. Hal pertama yang harus dilakukan oleh penjual dimana perlu mengakses *display inventory*. Pada *display inventory* terdapat satu *button* dengan nama *add item* dan apabila diberikan *action* akan memunculkan *form input produk* yang diberikan dengan fungsi modal. Pada

form tersebut dimana penjual akan melakukan *input* data produk yang dibutuhkan oleh sistem. Data yang di *input* kan tersebut akan dikirim kepada sistem. Sistem akan melakukan proses validasi guna mengecek data tersebut apakah sudah benar. Apabila data yang dikirim tidak ada masalah maka sistem akan melakukan verifikasi. Data yang sudah terverifikasi akan disimpan kedalam *database*. Pada *database* akan berjalan fungsi *PUT* yang dimana berguna untuk meng-*input*-kan data. Data yang sudah ter *input* akan di *upload* oleh *database* agar tampil pada *Display My Inventory*.

7) **Activity Diagram Tambah Produk Ke Cart**



Gambar 5.30 Activity Diagram Tambah Produk Ke Cart

Pada *activity* diagram tersebut dimana menjelaskan bagaimana aktifitas dalam melakukan penambahan data ke *cart* dan aktor yang berperan adalah *user* dengan level *pelanggan*. Hal pertama yang harus dilakukan pelanggan adalah mengakses *dashboard* terlebih dahulu. Dengan begitu pelanggan dapat memilih produk yang diinginkan. Pelanggan juga dapat melihat informasi – informasi produk tersebut dengan menekan *button* detail. Sistem akan memunculkan *display* detail produk. Apabila pelanggan merasa cocok dengan produk tersebut maka pelanggan dapat menambahkan produk tersebut kedalam *cart* dengan cara meng *click button add to cart*. Sistem akan secara otomatis menambahkan produk tersebut kedalam *cart*.