

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

Laporan ini dibuat untuk memenuhi persyaratan kelulusan  
Matakuliah Program Internship 2019/2020



**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

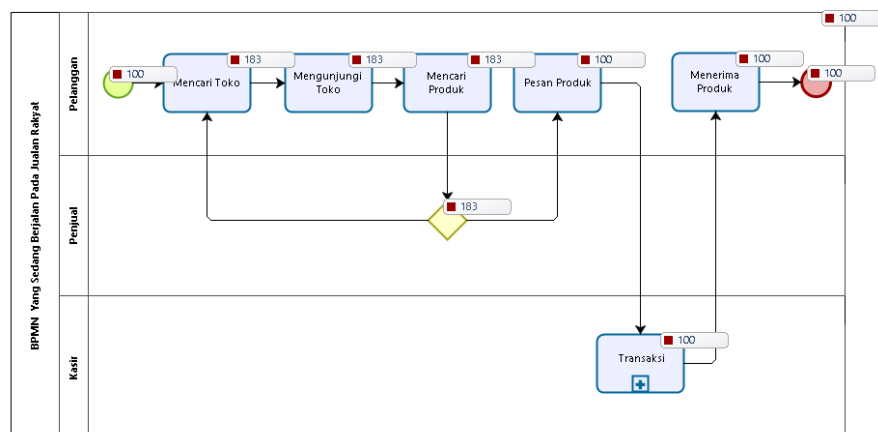
#### 5.1 Analisis Dan Perancangan Sistem

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 proses 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.

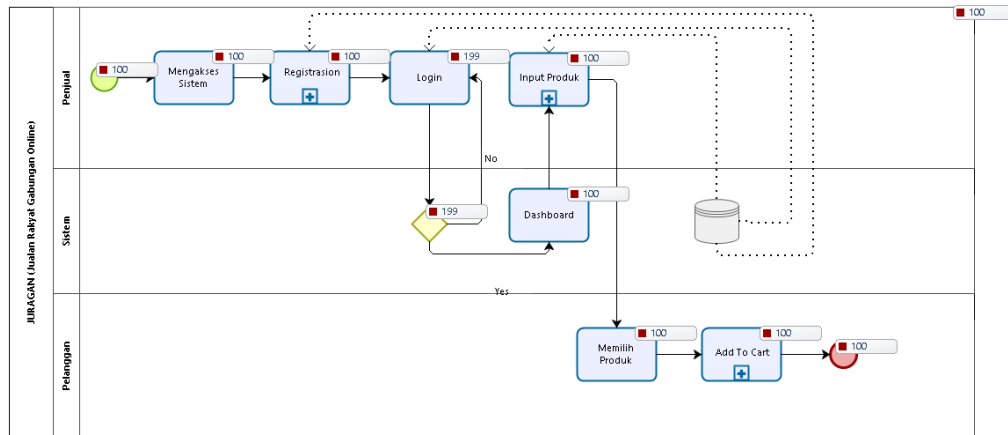
#### **5.1.2 Analisis Sistem Yang Akan Dibangun**

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



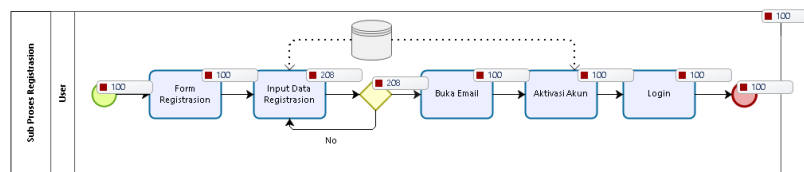
Gambar 5.2 BPMN Sistem JURAGAN

Tabel 5.2 Hasil Simulasi Dari BPMN Sistem JURAGAN

<i>Name</i>	<i>Type</i>	<i>Instances Completed</i>
Sistem JURAGAN	<i>Process</i>	100
<i>NoneStart</i>	<i>Start Event</i>	100
Mengakses Sistem	<i>Task</i>	100
<i>Login</i>	<i>Task</i>	199
<i>ExclusiveGateway</i>	<i>Gatewat</i>	199
<i>Dashboard</i>	<i>Task</i>	100
Memilih Produk	<i>Task</i>	100
<i>NoneEnd</i>	<i>End Event</i>	100
<i>Registration</i>	<i>Task</i>	100

Input Produk	<i>Task</i>	<i>100</i>
Add To Cart	<i>Task</i>	<i>100</i>

## 2) Analisis BPMN Pada Sub – Proses *Registration*

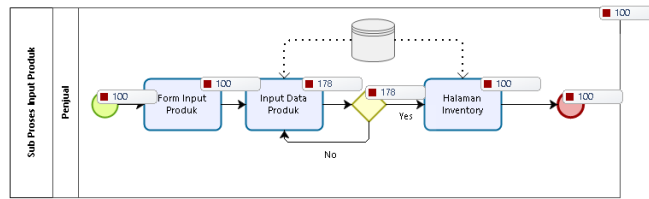


*Gambar 5.3 BPMN Sub – Proses Registrasion*

*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>
Buka Email	<i>Task</i>	<i>100</i>
Aktivasi Akun	<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

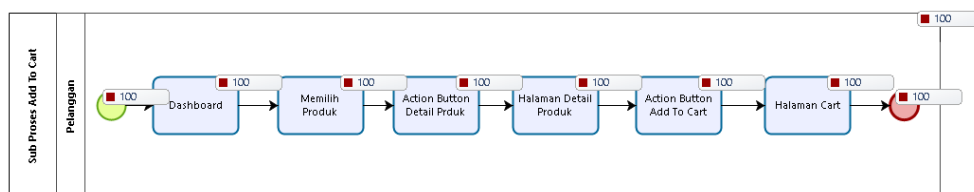


*Gambar 5.4 BPMN Sub – Proses Input Produk*

*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
<i>Halaman 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>	100

<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>
Action Button Detail Produk	<i>Task</i>	<i>100</i>
Halaman Detail Produk	<i>Task</i>	<i>100</i>
Action Button Add To Cart	<i>Task</i>	<i>100</i>
Halaman Cart	<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*

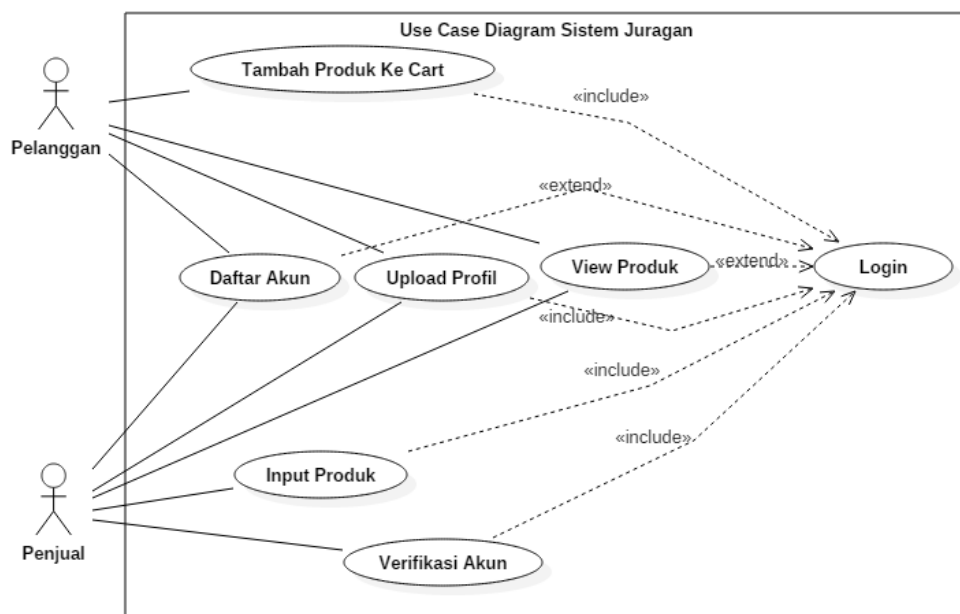
Dibuat Oleh	Penjual
-------------	---------

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	<ul style="list-style-type: none"> <li>a. Daftar Akun</li> <li>b. <i>Login</i></li> <li>c. <i>Upload</i> Profil</li> <li>d. <i>View</i> Produk</li> <li>e. Tambah Produk Ke <i>Cart</i></li> </ul>
2.	Penjual	<ul style="list-style-type: none"> <li>a. Daftar Akun</li> <li>b. <i>Login</i></li> <li>c. <i>Upload</i> Profil</li> <li>d. <i>View</i> Produk</li> <li>e. <i>Input</i> Produk</li> <li>f. Verifikasi Akun</li> </ul>

#### 5.1.2.3.1.2 Definisi Use Case

Tabel 5.9 Definisi Use Case

No.	Use Case	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</i> Produk	<ul style="list-style-type: none"> <li>a. Melakukan <i>view</i> terhadap produk yang sudah di <i>input</i> kan oleh penjual.</li> <li>b. Melihat detail produk secara keseluruhan.</li> </ul>

4.	<i>Upload Profil</i>	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>	a. Melakukan proses <i>input</i> berupa informasi – informasi produk secara detail oleh penjual.
6.	Tambah Produk Ke <i>Cart</i>	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</i> data <i>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 email</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 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 detail</i> pada produk yang pelanggan inginkan	a. <i>Display detail produk</i>

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 Profil Pelanggan</i>
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 edit 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 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
<b>Skenario</b>	
Kondisi Awal	<i>Display Change Password</i>
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
1. <i>Input password</i> lama	a. -
2. <i>Input password</i> baru	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

<b>Identifikasi</b>	
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
<b>Skenario</b>	



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 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

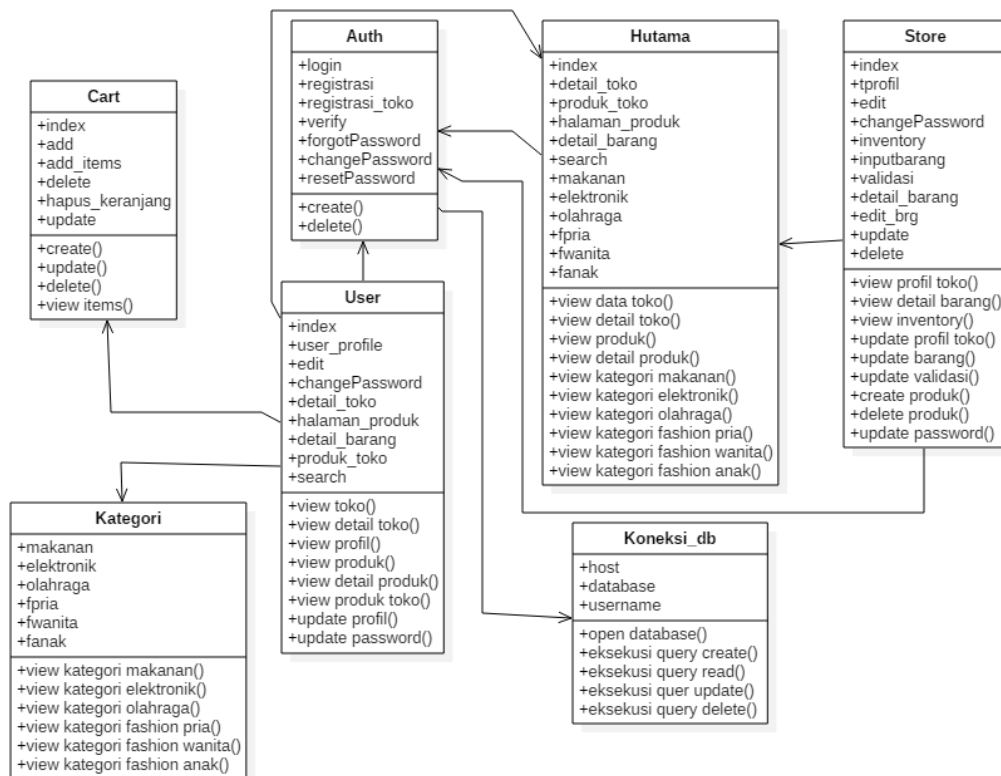
1. Memilih produk	a.
2. Melihat informasi produk	b. <i>Display</i> detail produk
3. Memberikan <i>action</i> pada <i>button add to cart</i>	c. Menambahkan produk ke dalam <i>cart</i>

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 :

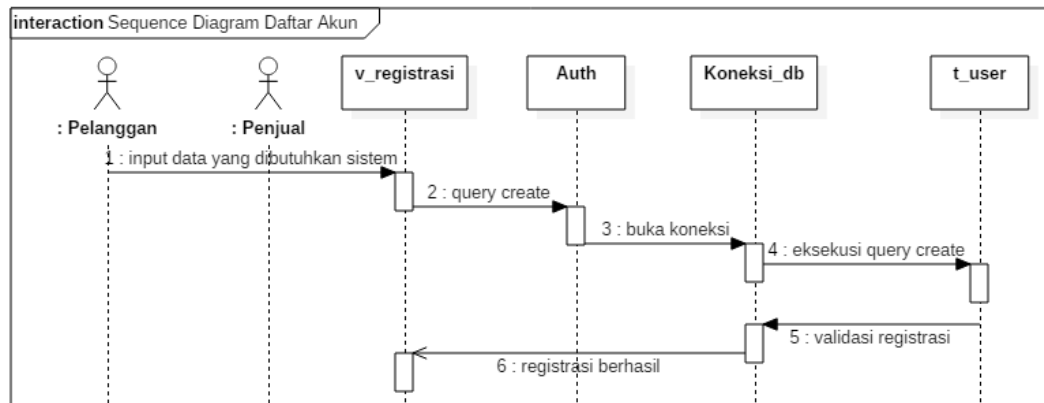


Gambar 5.7 Class Diagram Sistem JURAGAN

### 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

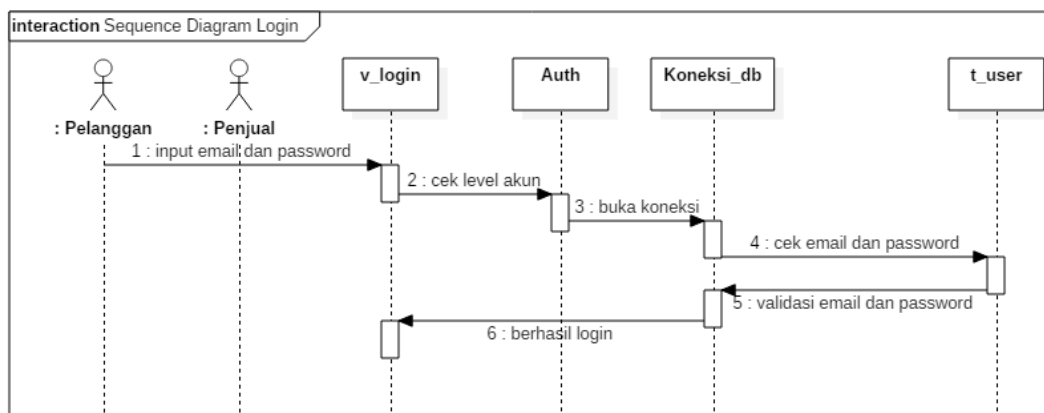


Gambar 5.8 Sequence Diagram Daftar Akun

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



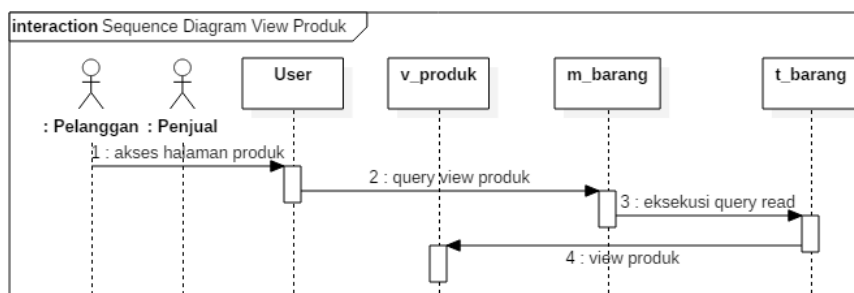
Gambar 5. 9 Sequence Diagram Login

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

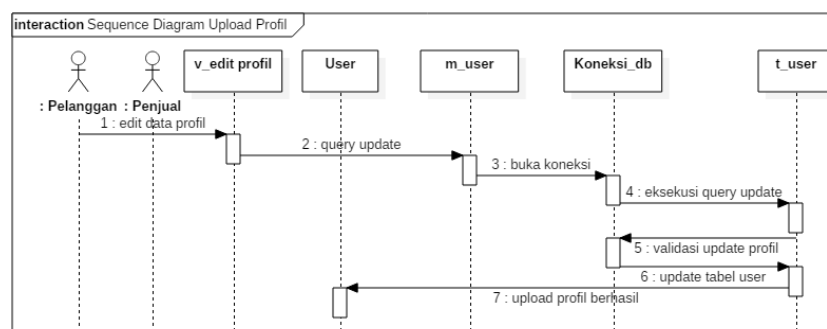


Gambar 5.10 Sequence Diagram View Produk Pelanggan

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

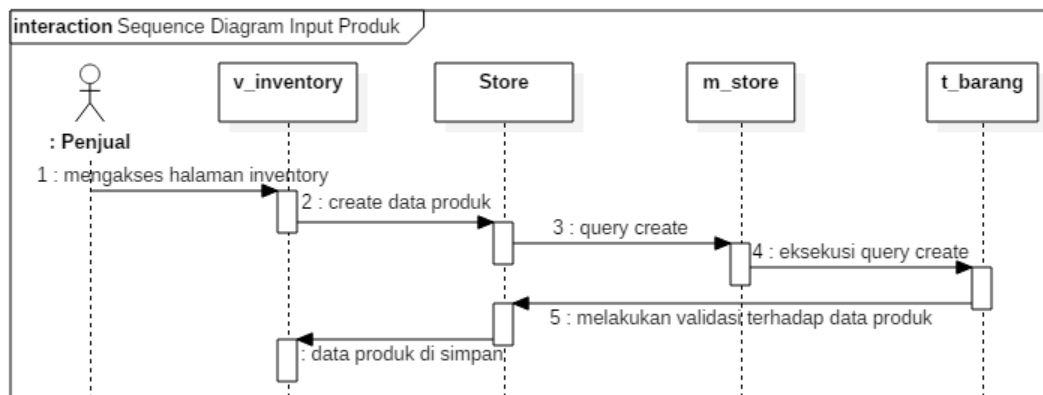


Gambar 5.11 Sequence Diagram Upload Profil

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



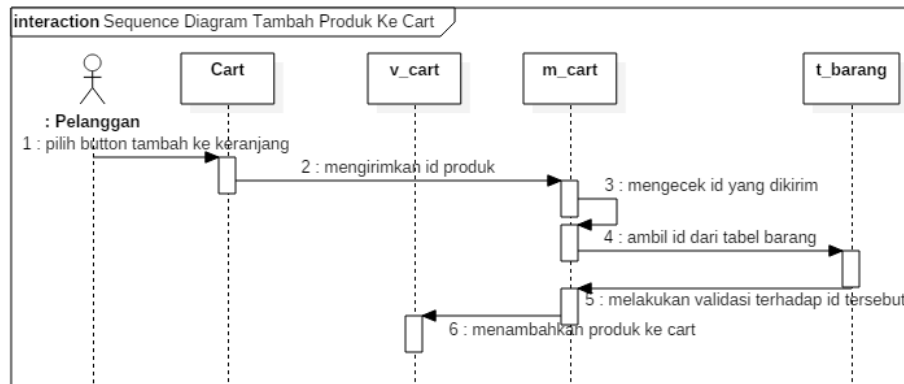
Gambar 5.12 Sequence Diagram Input Produk

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

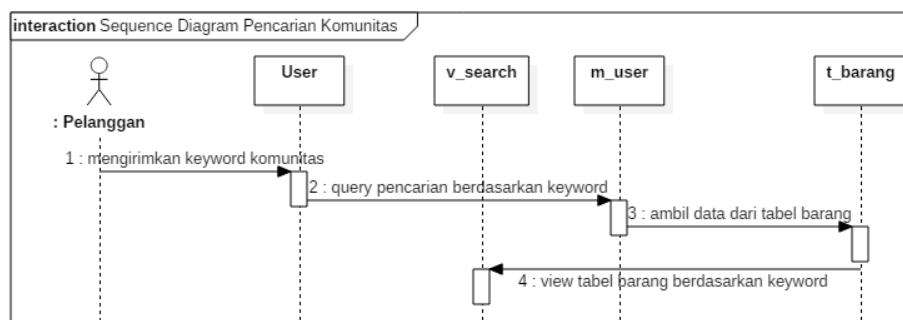


Gambar 5.13 Sequence Diagram Tambah Produk Ke Cart

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

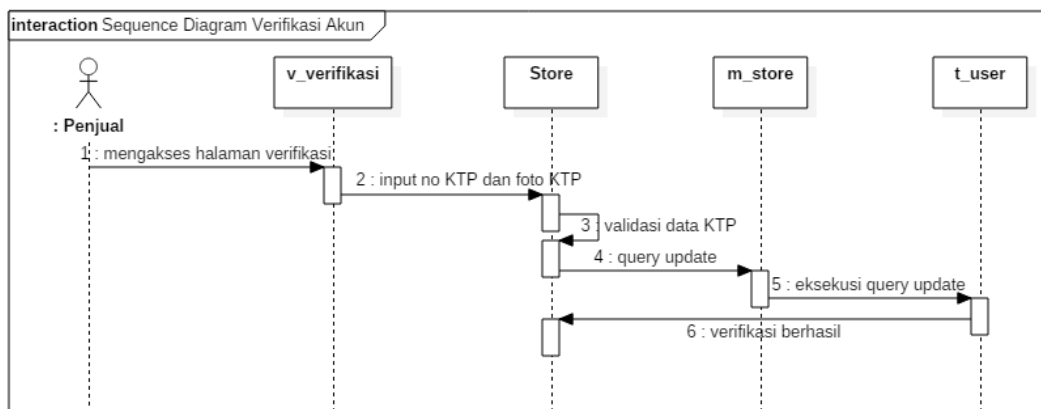


Gambar 5.14 Sequence Diagram Pencarian Komunitas

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 :

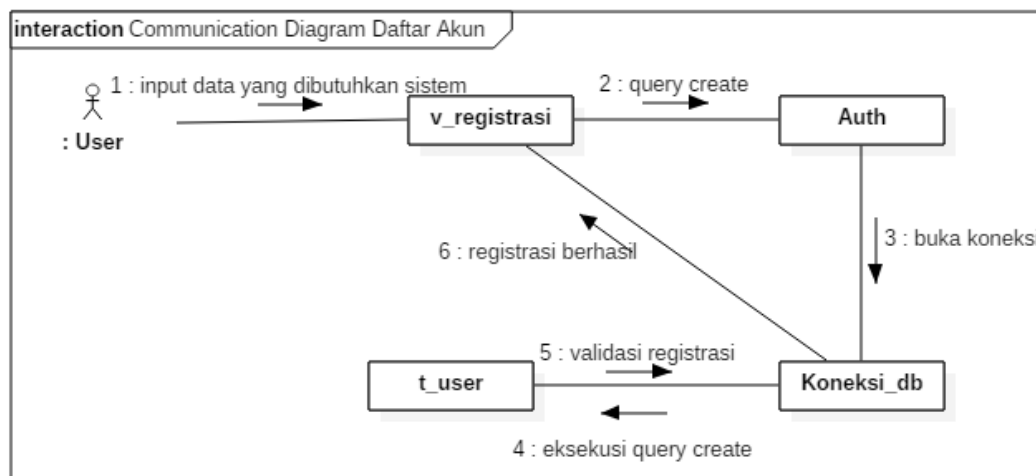
1. Penjual akan mengakses halaman verifikasi.
2. Pada halaman verifikasi dimana penjual perlu mengisi No KTP dan melakukan *upload* foto KTP.
3. Pada *controller store* dimana akan terjadinya proses validasi data KTP.
4. *Controller Store* akan mengirimkan *query update* kepada *model store*.
5. *Model 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

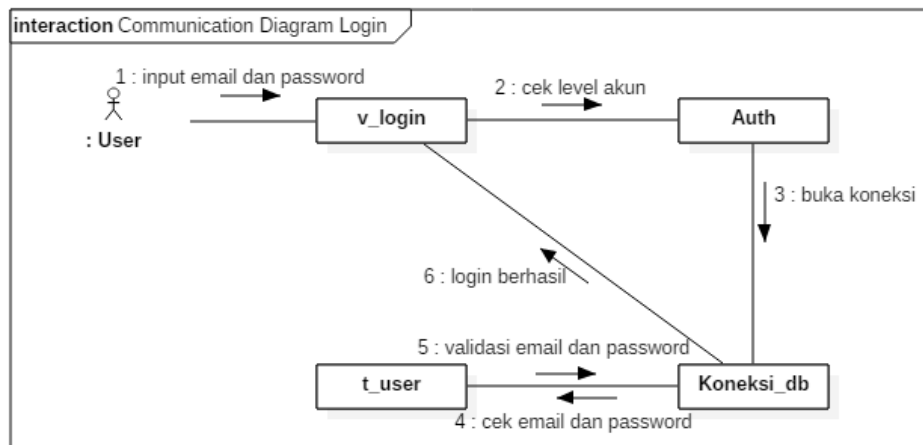


Gambar 5.16 *Collaboration* Diagram Daftar Akun

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

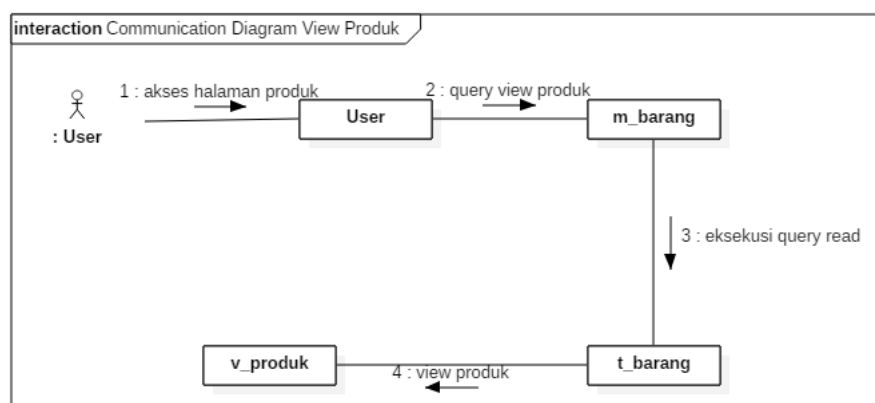


Gambar 5.17 Collaboration Diagram Login

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*.

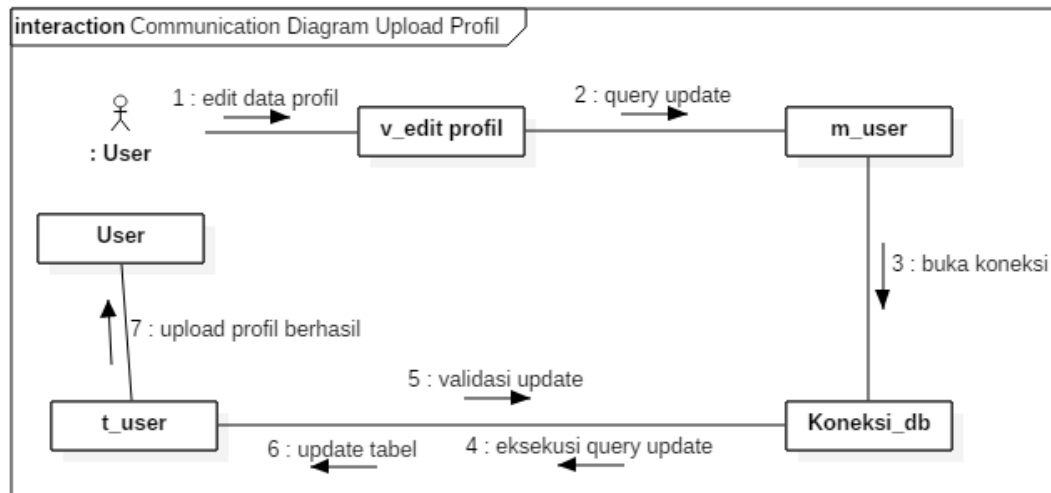
## 3) Collaboration Diagram View Produk



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**

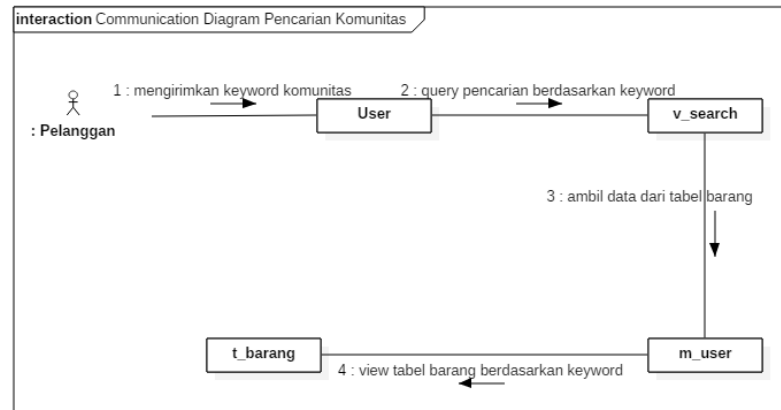


Gambar 5.19 Collaboration Diagram Upload Profil

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

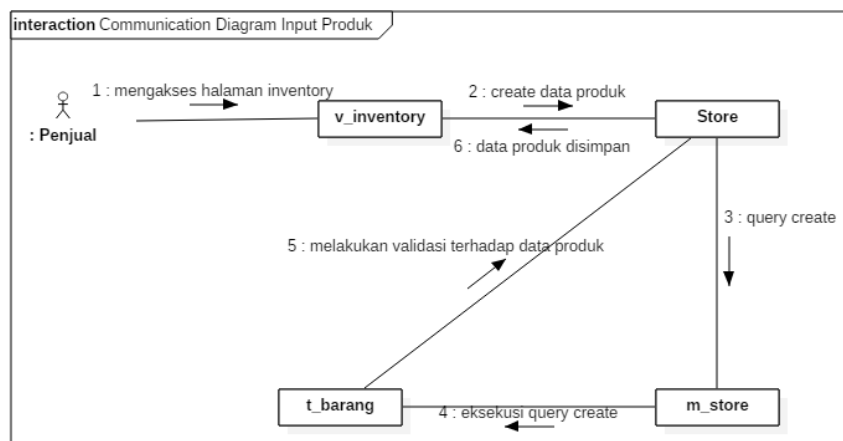


Gambar 5.20 Collaboration Diagram Pencarian Komunitas

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



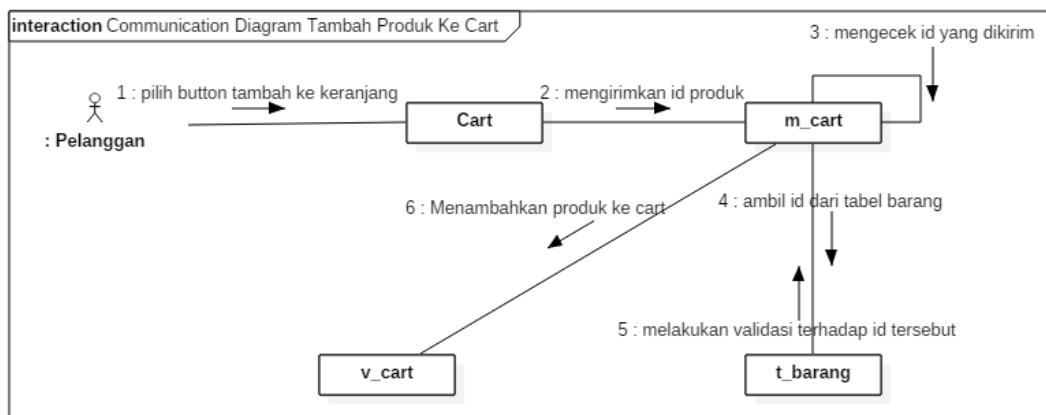
Gambar 5.21 Collaboration Diagram Input Produk

Keterangan :

1. Penjual akan melakukan akses terhadap halaman *inverntory*.

2. Pada halaman *inventory* tersebut pelanggan akan melakukan *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

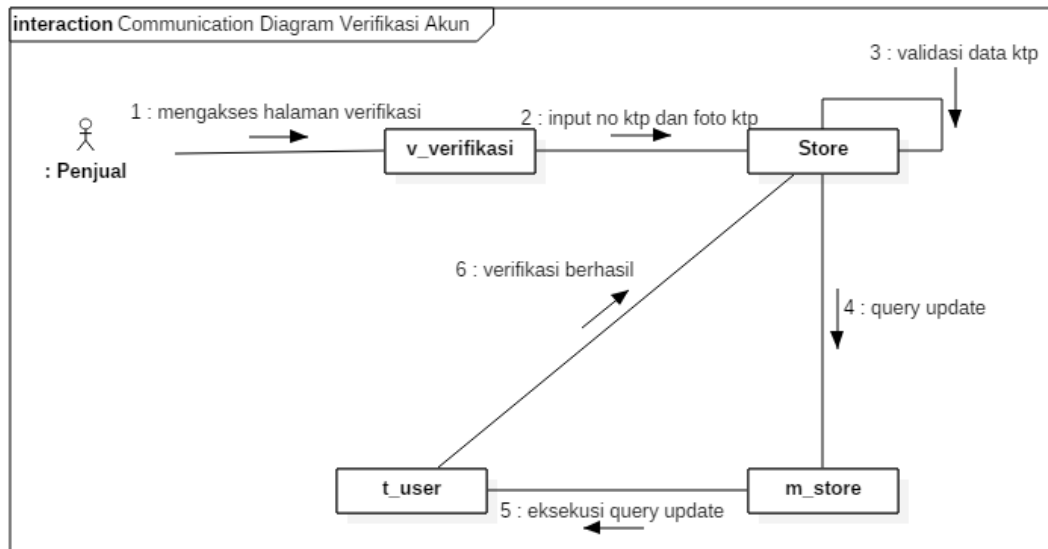


Gambar 5.22 Collaboration Diagram Tambah Produk Ke Cart

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

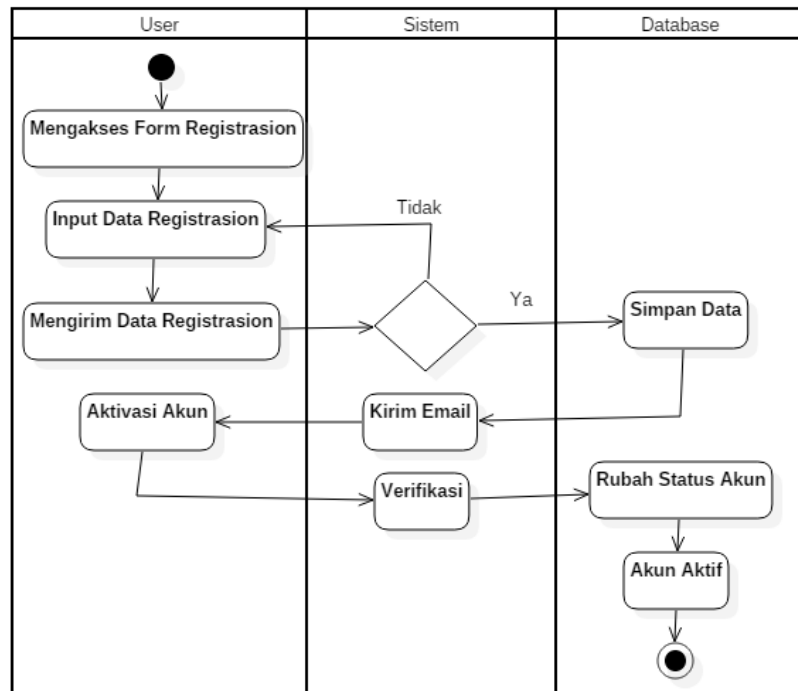
Keterangan :

1. Penjual akan mengakses halaman verifikasi.
2. Pada halaman verifikasi dimana penjual perlu 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.

### 5.1.2.3.5 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.

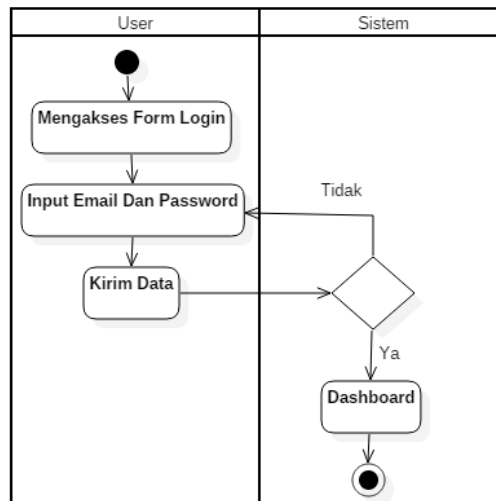
### 1) Activity Diagram Registrasion



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.

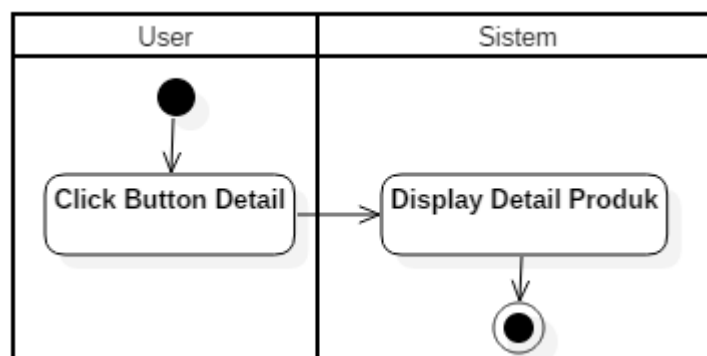
## 2) Activity Diagram Login



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

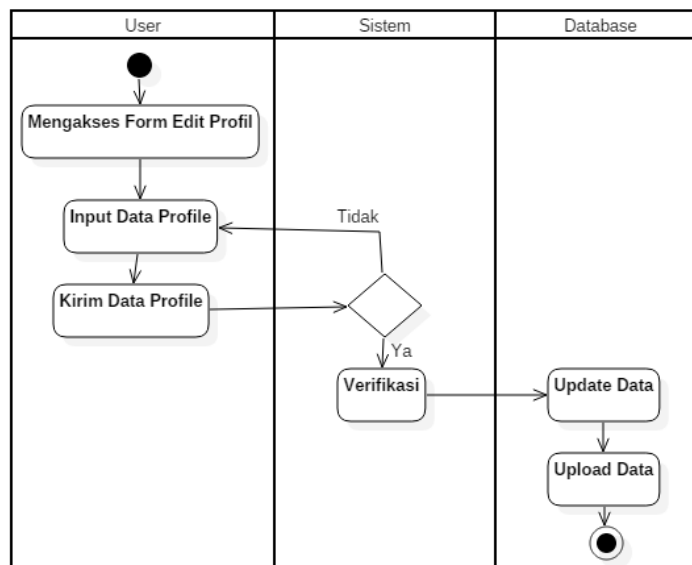


Gambar 5.26 Activity Diagram View Produk



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

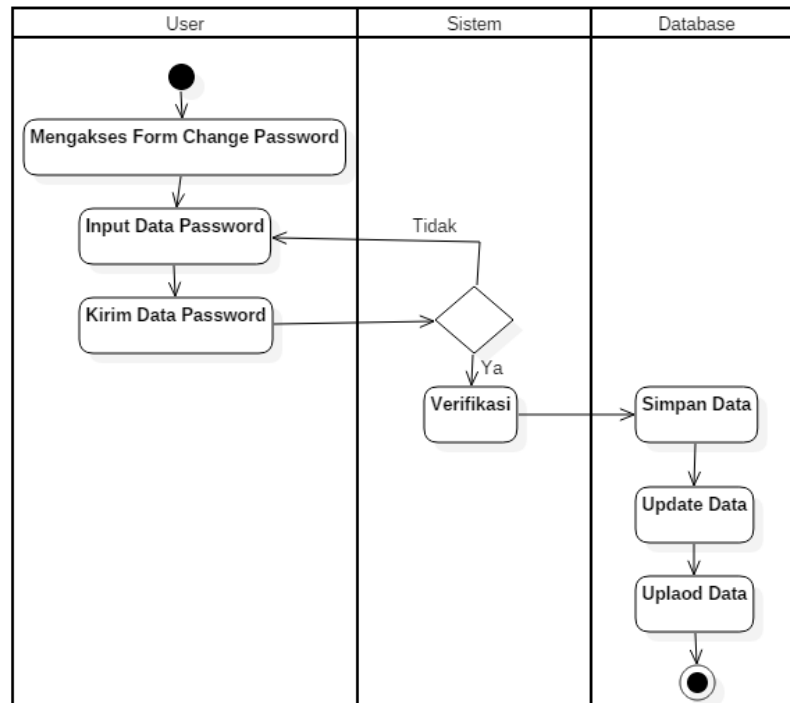


Gambar 5.27 Activity Diagram Edit Profile

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.

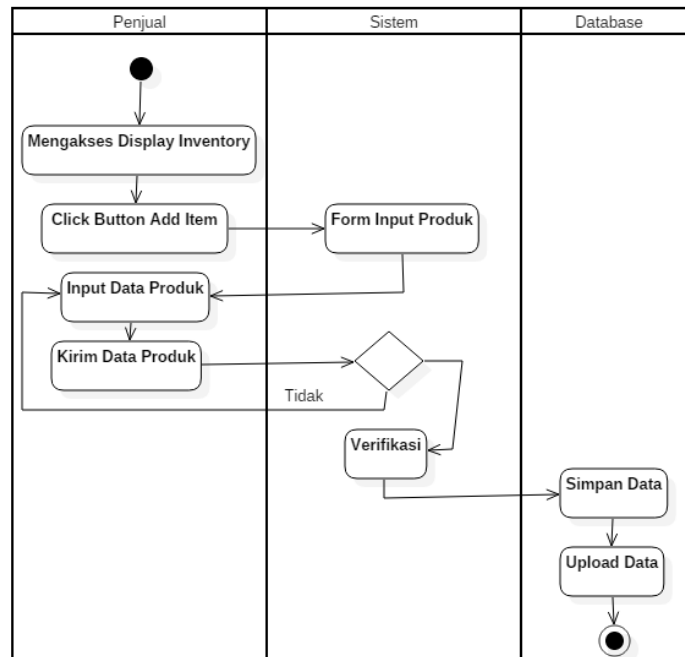
##### 5) Activity Diagram Change Password



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 diamana 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*.

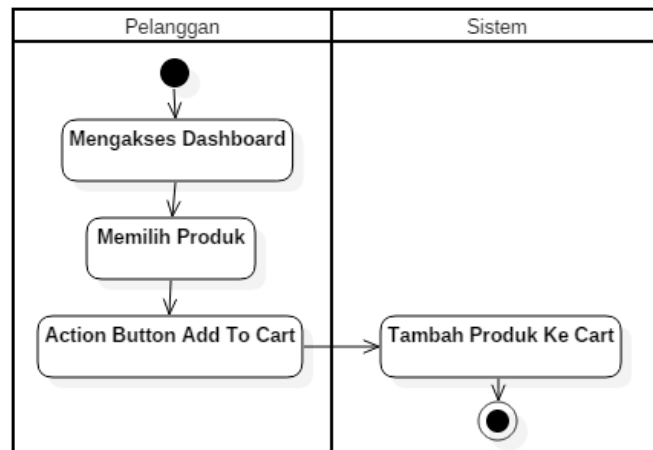
## 6) Activity Diagram *Input Produk*



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



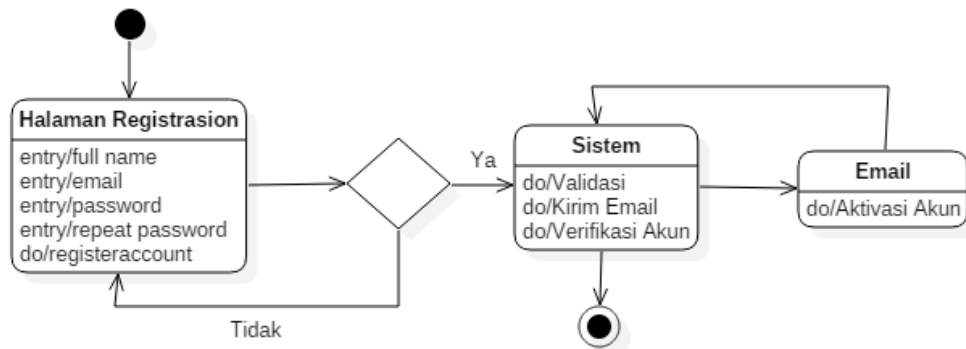
Gambar 5.30 Activity Diagram Tambah Produk Ke Cart

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*.

### 5.1.2.3.6 Statechart Diagram

*Statechart* diagram menggambarkan transisi dan perubahan keadaan (dari suatu *state* ke *state* lainnya) satu objek pada sistem sebagai akibat dari stimulasi yang diterima. *Statechart* diagram mendeskripsikan bagaimana suatu objek mengalami perubahan status adanya *trigger* dari *event-event*. Adapun *statechart* diagram pada sistem JURAGAN adalah sebagai berikut :

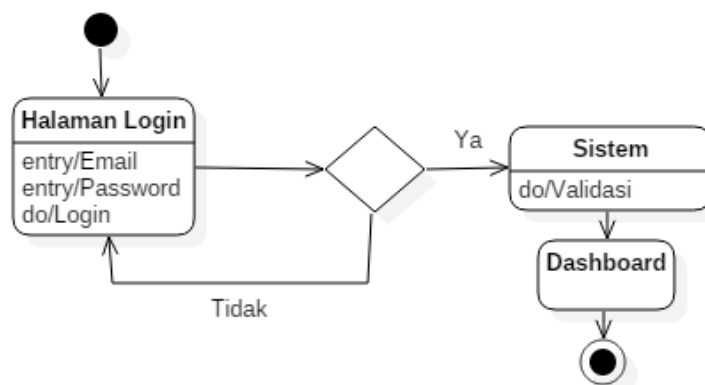
### 1) Statechart Diagram Registrasion



Gambar 5.31 Statechart Diagram Registrasion

Pada *statechart* diagram tersebut dimana menjelaskan *state – state* yang dilakukan oleh *user* untuk *registrasion* akun. *State* pertama yang perlu dilakukan adalah dengan mengakses halaman *registrasion* terlebih dahulu. Pada halaman *registrasion* dimana *user* perlu melakukan *input* data berupa *full name*, *email*, *password*, dan *repeat password*. Ketika data sudah di *input* kan maka *user* perlu memberikan *action* terhadap *button register accouunt*. *State* kedua dimana sistem akan melakukan proses validasi dan mengirimkan email kepada *user*. *State* ketiga dimana *user* akan mengakses *email* dan melakukan aktivasi akun. *State* keempat dimana sistem akan melakukan verifikasi akun sehingga akun menjadi aktif.

### 2) Statechart Diagram Login



Gambar 5.32 Statechart Diagram Login

Pada *statechart* diagram tersebut dimana menjelaskan *state – state* yang perlu dilakukan dalam proses *login*. Dimana untuk melakukan *login state* pertama ialah dengan mengakses halaman *login* terlebih dahulu. Pada halaman *login* dimana *user* perlu melakukan *input email* dan *password* yang sudah terdaftar pada sistem. Ketika data sudah di *input* maka *user* perlu memberikan *action* terhadap *button login*. *State* kedua dimana sistem akan melakukan validasi guna melakukan pengecekan terhadap data yang dikirim. Apabila data yang dikirim tidak terjadi masalah maka sistem akan melanjutkan ke proses berikutnya. *State* ketiga dimana sistem akan memunculkan *dashboard* yang menunjukkan bahwa *login* telah sukses.

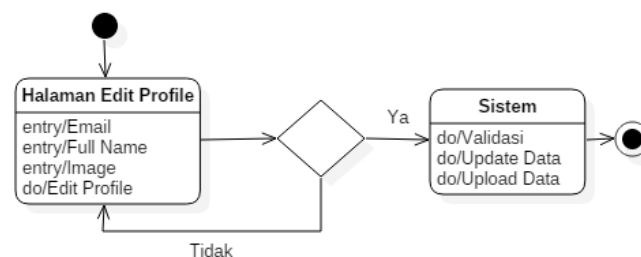
### 3) *Statechart* Diagram View Produk



Gambar 5.33 *Statechart* Diagram View Produk

Pada *statechart* diagram tersebut dimana menunjukkan *state – state* dalam melakukan *view* produk. *State* pertama yang perlu dilakukan adalah dengan memberikan *action* terhadap *button detail*. *State* kedua yaitu dimana sistem akan memberikan respon dengan cara memunculkan halaman detail produk.

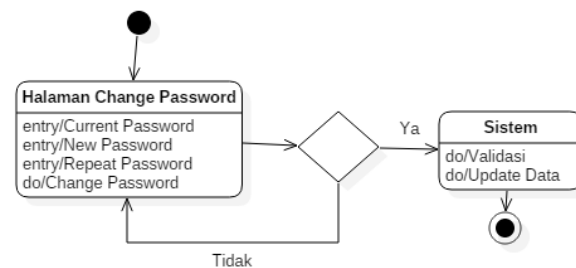
### 4) *Statechart* Diagram Edit Profile



Gambar 5.34 *Statechart* Diagram Edit Profile

Pada *statechart* diagram tersebut dimana menunjukkan *state* – *state* dalam melakukan *edit profile*. *State* pertama yang perlu dilakukan *user* adalah dengan mengakses halaman *edit profile*. Dalam halaman *edit profile* tersebut dimana *user* perlu menginputkan data *full name* dan *image*. *Email* disini bersifat *read only* sehingga tidak dapat diubah. Jika data tersebut sudah di *input* kan dimana *user* perlu memberikan *action* terhadap *button edit profile*. *State* kedua dimana sistem akan melakukan proses validasi guna mengecek data apakah sudah sesuai dengan ketentuan sistem. Apabila data sudah sesuai maka data akan dikirim ke dalam *database* untuk di *update*. Ketika sudah di *update* maka data tersebut akan di *upload* agar dapat dilihat perubahannya oleh *user*.

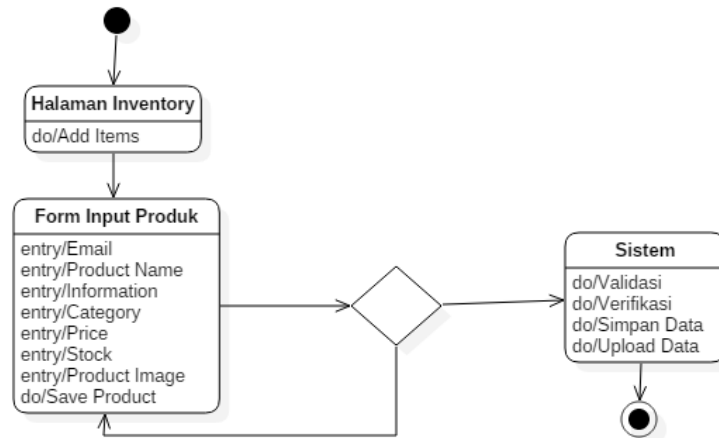
##### 5) *Statechart Diagram Change Password*



Gambar 5.35 *Statechart Diagram Change Password*

Pada *statechart* diagram tersebut dimana menunjukkan *state* – *state* dalam melakukan *change password*. *State* pertama yang perlu dilakukan adalah dengan mengakses halaman *change password*. Pada halaman tersebut dimana *user* perlu melakukan *input* data *current password*, *new password*, dan *repeat password*. Ketika data sudah di *input* kan dimana *user* perlu memberikan *action* terhadap *button change password*. *State* kedua yaitu dimana sistem akan melakukan proses validasi guna mengecek data apakah sudah sesuai dengan ketentuan sistem. Apabila data sudah sesuai maka sistem akan mengirim data tersebut ke dalam *database*. *Database* akan melakukan proses *get* dimana berfungsi untuk *update* data.

## 6) Statechart Diagram Input Produk

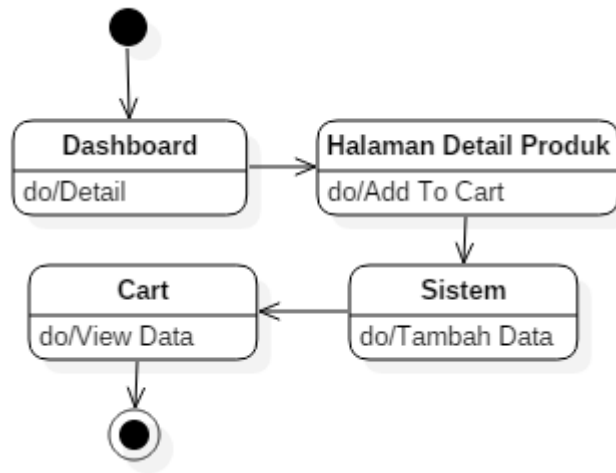


Gambar 5.36 Statechart Diagram Input Produk

Pada *statechart* diagram tersebut merupakan *state – state* dalam melakukan proses *input produk* dimana aktor yang berperan adalah *user* dengan level penjual. *State* pertama yang perlu dilakukan adalah dengan mengakses halaman *inventory* terlebih dahulu, dimana pada halaman tersebut terdapat satu *button add item* yang perlu di berikan *action*. *State* kedua yaitu sebuah modal berupa *form input product* akan muncul dan penjual perlu melakukan *input* data berupa *email*, *product name*, *information*, *category*, *price*, *stock*, dan *product image*. Ketika data sudah di *inputkan* dimana penjual perlu memberikan *ation* terhadap *button save produt*. *State* ketiga dimana sistem akan melakukan proses validasi guna mengecek data apakah sudah sesuai. Apabila data sudah sesuai maka sistem akan melakukan verifikasi data tersebut dan menyimpannya kedalam *database*. Pada *database* dimana akan berjalan fungsi *PUT* yang berperan dalam meng *input* kan data dan data yang sudah ter *input* akan di *upload* oleh *database*.



### 7) Statechart Diagram Tambah Produk Ke Cart

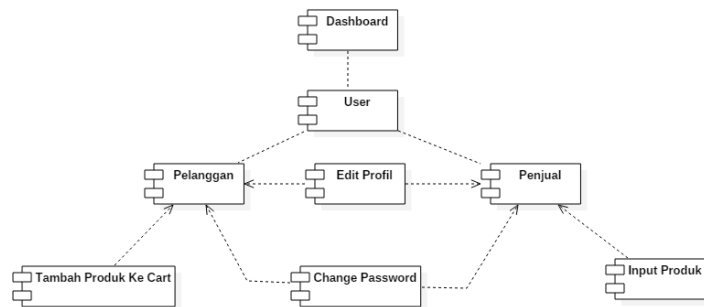


Gambar 5.37 Statechart Diagram Tambah Produk Ke Cart

Pada *statechart* diagram tersebut dimana akan menjelaskan *state* – *state* yang dilalui dalam proses menambahkan produk ke dalam *cart* yang dimana aktor disini adalah *user* dengan level pelanggan. *State* pertama dimana pelanggan perlu mengakses *dashboard* terlebih dahulu. Pada *dashboard* pelanggan dapat memilih produk – produk yang telah di *input* kan oleh penjual. Untuk melihat *informasion* produk tersebut dimana pelanggan dapat memberikan *action* terhadap *button* detail. *State* kedua dimana sistem akan merespon dengan memunculkan halaman detail produk. Pada halaman detail produk dimana pelanggan dapat menambahkan produk ke *cart* secara langsung dengan cara memberikan *action* terhadap *button* *add to cart*. *State* ketiga dimana sistem akan melakukan proses tambah data ke dalam halaman *cart*. *State* keempat dimana produk yang sudah ditambahkan dapat di *view* pada halaman *cart*.

#### 5.1.2.3.7 Component Diagram

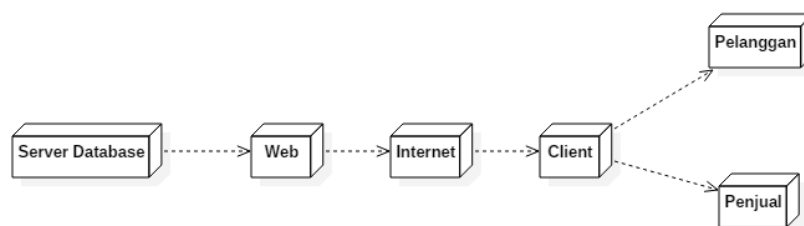
Berikut ini merupakan *component* diagram yang menjelaskan semua *component* yang terdapat pada sistes JURAGAN.



Gambar 5.38 Component Diagram

#### 5.1.2.3.8 Diployment Diagram

*Deployment* diagram menunjukkan tata letak sebuah sistem secara fisik, menampilkan bagian – bagian *software* yang berjalan pada bagian – bagian *hardware* yang digunakan untuk mengimplementasikan sebuah sistem dan keterhubungan antara komponen – komponen *hardware* tersebut. *Deployment* diagram dapat digunakan pada bagian – bagian awal proses perancangan sistem untuk mendokumentasikan arsitektur fisik sebuah sistem. Berikut *deployment* diagram yang menggambarkan susunan fisik, perangkat lunak dan sistem.



Gambar 5.39 Deployment Diagram

## 5.2 Perancangan Basis Data /Database Sistem

Dalam pembuatan sistem JURAGAN (Jualan Rakyat Gabungan Online) dimana menggunakan *database* MySQL dengan nama “juragan” dan tersusun menjadi beberapa tabel seperti berikut :

Tabel 5.20 User

No.	Field	Type	Wide	Information
1	<i>Id</i>	<i>Integer</i>	<i>11</i>	<i>Id user yang merupakan primary key</i>
2	<i>Name</i>	<i>Varchar</i>	<i>128</i>	Nama lengkap <i>user</i>
3	<i>Email</i>	<i>Varchar</i>	<i>128</i>	<i>Email dari user</i>
4	<i>Password</i>	<i>Varchar</i>	<i>300</i>	<i>Password dari akun user</i>
5	<i>Image</i>	<i>Varchar</i>	<i>128</i>	<i>Image profile pada user pelanggan</i>
6	<i>Role_id</i>	<i>Integer</i>	<i>11</i>	Merupakan id level <i>user</i> .  1. Admin 2. User
7	<i>Role_toko</i>	<i>Integer</i>	<i>11</i>	Merupakan id toko yang sudah mendaftar.  0. Belum Mendaftar 1. Sudah Mendaftar
8	<i>Is_active</i>	<i>Integer</i>	<i>11</i>	Id akun yang aktif.  0. Belum Aktif 1. Aktif
9	<i>Date_created</i>	<i>Integer</i>	<i>11</i>	Tanggal <i>user</i> mendaftar akun

10	<i>Name_store</i>	<i>Varchar</i>	<i>128</i>	Nama dari akun penjual
11	Alamat	<i>Varchar</i>	<i>600</i>	Alamat dari <i>user</i>
12	Alamat_toko	<i>Varchar</i>	<i>600</i>	Alamat toko tersebut
13	Jenis_usaha	<i>Varchar</i>	<i>25</i>	Jenis Usaha pada toko tersebut. (UMKM, Pengusaha Mikro, Pedagang Kaki Lima).
14	Tlp_toko	<i>Varchar</i>	<i>15</i>	No telpon yang aktif pada toko tersebut.
15	<i>Image_store</i>	<i>Varchar</i>	<i>128</i>	<i>Image</i> dari akun penjual

Tabel 5.21 *tb\_barang*

<b>No.</b>	<b><i>Field</i></b>	<b><i>Type</i></b>	<b><i>Wide</i></b>	<b><i>Information</i></b>
1	Id_brg	<i>Integer</i>	<i>11</i>	Merupakan id dari barang dan merupakan <i>primary key</i>
2	<i>Email</i>	<i>Varchar</i>	<i>128</i>	<i>Forgent key</i> dari tabel <i>user</i>
3	Nama_brg	<i>Varchar</i>	<i>120</i>	Nama dari produk yang di <i>input</i>
4	Keterangan	<i>Varchar</i>	<i>225</i>	Keterangan dari sebuah produk
5	Kategori	<i>Varchar</i>	<i>60</i>	Kategori dari sebuah produk
6	Harga	<i>Integer</i>	<i>11</i>	Harga dari sebuah produk

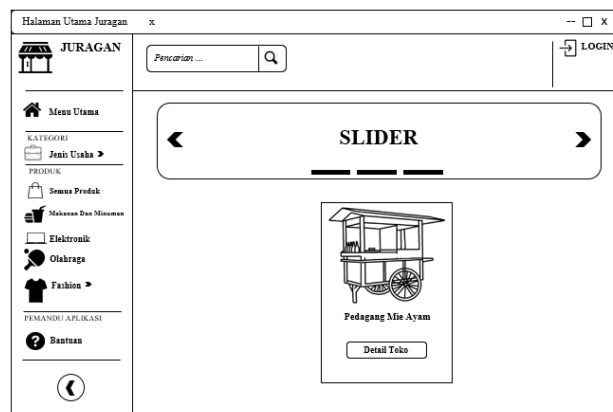
7	Stok	<i>Integer</i>	4	Merupakan jumlah dari sebuah produk yang ada
8	Gambar	<i>Varchar</i>	60	Merupakan gambar dari sebuah produk

Tabel 5. 22 User\_token

No.	Field	Type	Wide	Information
1	<i>Id</i>	<i>Integer</i>	11	Merupakan <i>id</i> dari token yang masuk dan sebuah <i>primary key</i>
2	<i>Email</i>	<i>Varchar</i>	128	Merupakan <i>forgent key</i> dari tabel <i>user</i>
3	<i>Token</i>	<i>Varchar</i>	128	Token yang akan dikirim saat aktivasi akun
4	<i>Date_created</i>	<i>Varchar</i>	128	Tanggal token di <i>input</i> kan

### 5.3 Perancangan User Interface Sistem

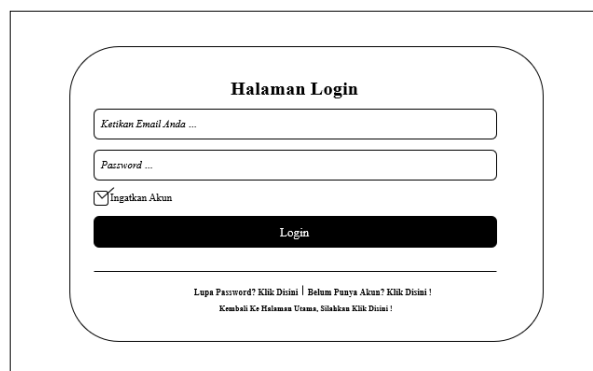
#### 1). User Interface Halaman Utama (Sebelum Login)



Gambar 5.40 UI Halaman Utama Sebelum Login

Pada gambar 5.40 dimana merupakan UI halaman utama dengan kondisi sebelum melakukan *login*. Dalam *topbar* pada UI tersebut terdapat sebuah *form* pencarian yang berguna untuk mencari sebuah toko. Terdapat juga sebuah *icon login* yang disertai dengan *text* dengan fungsi *link*, dan apabila di klik akan menuju ke halaman *login*. Pada bagian *sidebar* terdapat *icon-icon* yang dibutuhkan dalam sistem JURAGAN tersebut yang diberikan dengan fungsi *link*. Pada bagian *index* nya terdapat *slider* yang akan berpindah – pindah secara otomatis. Terdapat juga *card* yang di isi dengan toko – toko yang sudah terdaftar pada sistem JURAGAN tersebut.

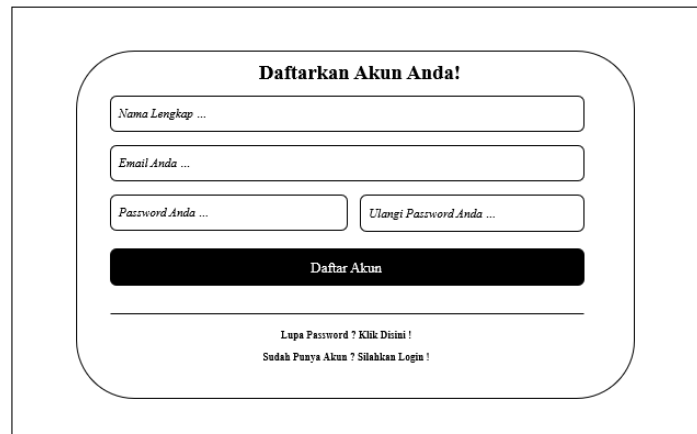
## 2). User Interface Login

The image shows a login form titled "Halaman Login". It contains two input fields: "Ketikkan Email Anda ..." and "Password ...". Below these is a checkbox labeled "Ingatkan Akun". A black button labeled "Login" is positioned below the checkbox. At the bottom of the form, there is a line of text: "Lupa Password? Klik Disini | Belum Punya Akun? Klik Disini | Kembali Ke Halaman Utama, Silahkan Klik Disini!".

Gambar 5.41 UI Login

Pada gambar 5.41 tersebut merupakan UI *login* dari sistem JURAGAN. Dimana UI *login* tersebut di masukkan kedalam *card* dan diberikan sebuah *form* guna menginputkan *email* yang terdaftar dan *password*. Terdapat juga pengingat akun untuk melakukan save pada akun kita. Terdapat juga *buuton login* yang dimana apabila diklik akan memulai proses *login*.

### 3). *User Interface* Daftar Akun

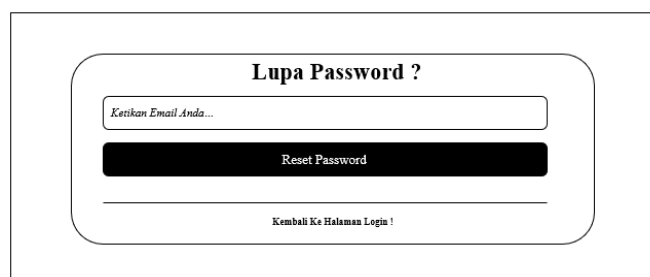


The image shows a registration form titled "Daftarkan Akun Anda!". It contains four input fields: "Nama Lengkap ..." (Full Name), "Email Anda ..." (Your Email), "Password Anda ..." (Your Password), and "Ulangi Password Anda ..." (Repeat Your Password). Below these fields is a large black button labeled "Daftar Akun". At the bottom of the form, there are two links: "Lupa Password ? Klik Diini !" (Forgot Password? Click Here!) and "Sudah Punya Akun ? Silahkan Login !" (Already have an account? Please login!).

*Gambar 5.42 UI Daftar Akun*

Pada *UI* daftar akun tersebut dimana hanya terdapat satu *layer* yang didalamnya terdiri dari 4 *textbox* yang diberikan fungsi *placeholder*, terdapat juga 1 *button* untuk melakukan *action registrasion*. Di bagian bawah *layer* tersebut terdapat 2 *link* yang terdiri dari *Forgot Password* dan *Already have as account ? Login !* yang dimana akan menuju ke halaman lain sesuai *link* tersebut apabila di *click*.

### 4). *User Interface* Lupa Password

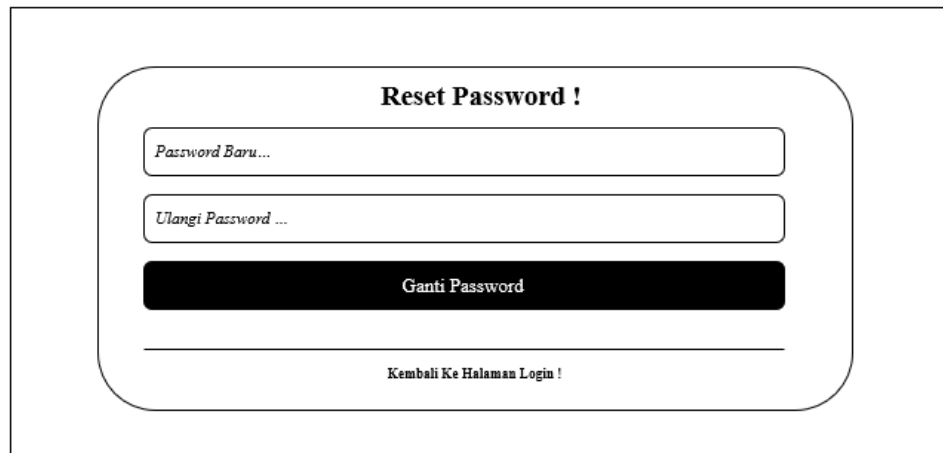


The image shows a "Forgot Password" form titled "Lupa Password ?". It contains one input field labeled "Ketikkan Email Anda..." (Type your email...). Below the input field is a large black button labeled "Reset Password". At the bottom of the form, there is a link: "Kembali Ke Halaman Login !" (Back to Login Page!).

*Gambar 5.43 UI Forgot Password*

Pada *UI Forgot Password* dimana hanya terdiri dari satu *layer* utama yang di dalamnya terdapat 1 *textbox* dengan menggunakan fungsi *placeholder*, terdapat juga 1 *button* untuk melakukan *action reset password*. Pada bagian bawah *layer* terdapat satu *link Back to login* yang dimana akan berpindah ke halaman *login* apabila di *click*.

### 5). User Interface Reset Password

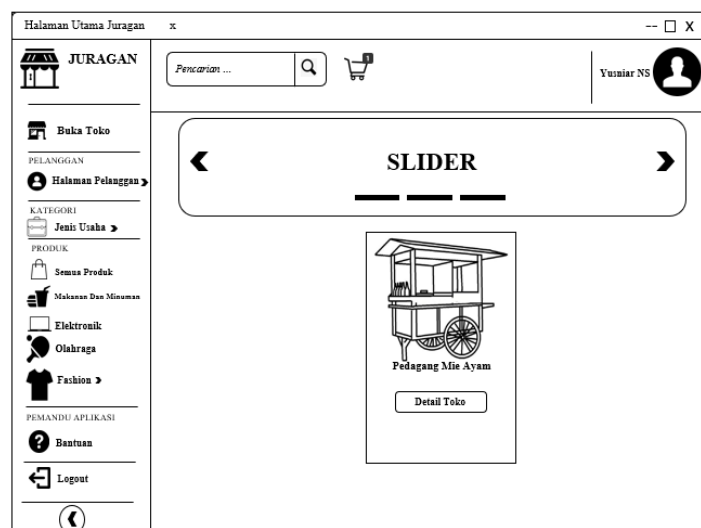


The image shows a 'Reset Password' form. It has a title 'Reset Password !' at the top. Below the title are two text input fields: 'Password Baru...' and 'Ulangi Password ...'. Below these fields is a black button with the text 'Ganti Password'. At the bottom of the form is a link that says 'Kembali Ke Halaman Login !'.

Gambar 5.44 UI Reset Password

Pada UI Reset Password tersebut hanya terdapat satu *layer* utama yang dimana di dalamnya terdapat 2 *textbox* dengan fungsi *placeholder*, terdapat juga 1 *button* untuk melakukan *action changed password*. Di bagian bawah *layer* terdapat 1 *link* untuk menuju ke halaman *login* apabila di *click*.

### 6). User Interface Halaman Utama Sesudah Login



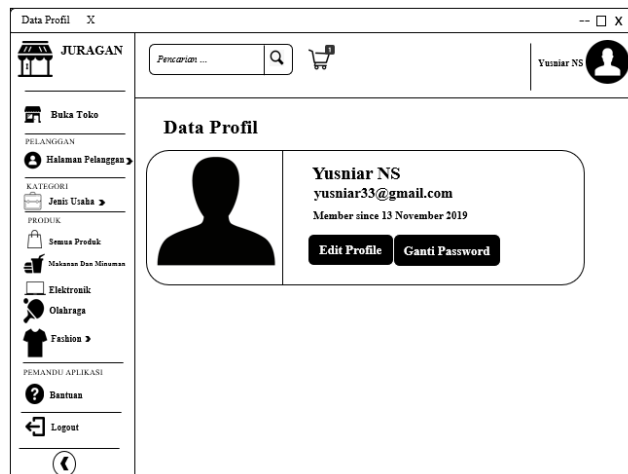
Gambar 5.45 UI Halaman Utama Sesudah Login

Pada gambar 4.45 merupakan UI pada halaman utama sistem JURAGAN dengan kondisi sesudah *login*, dimana bedanya dengan gambar 5.40 UI pada



gambar 4.45 ini memberikan akses yang lebih luas yaitu dengan adanya *icon-icon* yang bertambah seperti buka toko dan halaman pelanggan pada *sidebar* dan juga nama *user* beserta foto profil dan *icon cart* pada *topbar*.

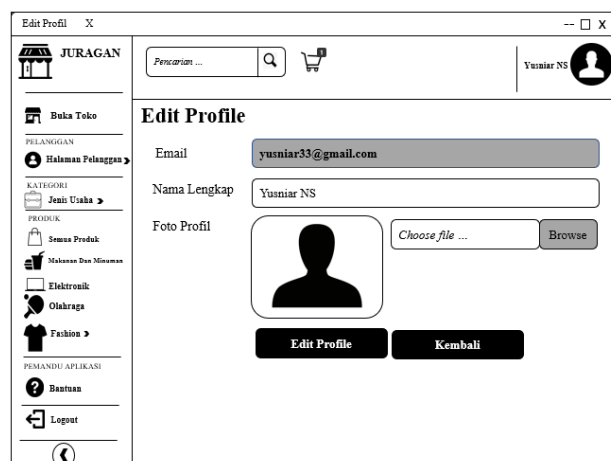
## 7). User Interface Data Profil



Gambar 5.46 UI Data Profil

Pada gambar 5.46 merupakan UI dari data profil dimana untuk bagian *sidebar* dan *topbar* akan terlihat sama dengan gambar 5.45. Pada bagian index nya terdapatsebuah *card* yang didalamnya diberikan informasi – informasi data profil *user*.

## 8). User Interface Edit Profil



Gambar 5.47 UI Edit Profil

Pada gambar 5.47 merupakan UI dari edit profil pelanggan dimana pada bagian *sidebar* dan *topbar* terlihat sama dengan gambar 5.45. Pada bagian *index* dimana terdapat sebuah *form* yang pada setiap *textbox* nya di ambil dari *database*. Pada bagian *email* dimana bersifat *readonly*. Terdapat dua buah *button* sebagai memulai aksi yang diberikan.

### 9). User Interface Ganti Password

Gambar 5.48 UI Ganti Password

Pada gambar 5.48 tersebut dimana merupakan UI dari ganti password bagi *user*. Pada bagian *sidebar* dan *topbar* akan terlihat sama dengan gambar 5.45. Pada bagian *index* terdapat sebuah *form* yang diberikan 3 *textbox* dan 2 *button* untuk memulai aksi ganti *password*.

## 10). User Interface Profil Toko

Gambar 5.49 UI Profil Toko

Pada gambar 5.49 merupakan UI dari profil toko bagi *user* penjual. Pada bagian *sidebar* terdapat beberapa *icon* yang nampak lebih sedikit dari pada gambar 5.45. Hal ini dikarenakan bedanya akses yang diberikan. Pada bagian *topbar* sekilas terlihat sama dengan gambar 5.45 akan tetapi yang membedakan adalah tidak adanya *icon cart* yang dimana *icon cart* merupakan sesuatu *fitur* yang hanya diberikan oleh *user* dengan level pelanggan. Pada bagian *index* nya terdapat satu *form* yang diberikan beberapa *textbox* serta *button edit* profil toko yang dimana akan melakukan pendaftaran pada toko tersebut.

## 11). User Interface Produk Toko

No.	Nama Barang	Keterangan	Stok	Harga	Aksi
1	Laptop Asus X 555 U	Intel core i5 - 6200	10	Rp. 6.400.000	
2	Samsung Galaxy S10	8 GB / 128 GB	20	Rp. 10.000.000	
3	Canon 80D Kit EF-S	24,2 MP APS-C	15	Rp. 17.890.000	

Gambar 5.50 UI Produk Toko

Pada gambar 5.50 merupakan UI dari produk toko yang dimana akses tersebut diberikan kepada *user* dengan level pelanggan. Pada bagian *sidebar* dan *topbar* akan terlihat sama dengan gambar 5.49. Pada bagian *index* nya dimana terdapat tabel untuk melakukan *view* dari *database* berdasarkan *session* yang diberikan. Terdapat juga satu *button* yang berfungsi untuk mengakses halaman *input* produk.

## 12). User Interface Form Input Produk

The image shows a web form titled "FORM INPUT PRODUK". It contains several input fields and two buttons at the bottom. The fields are labeled as follows:

- Email: Contains the text "yusniar33@gmail.com".
- Nama Produk: An empty text input field.
- Detail Produk: An empty text input field.
- Kategori: An empty text input field.
- Harga: An empty text input field.
- Stok: An empty text input field.
- Gambar Produk: An empty text input field.

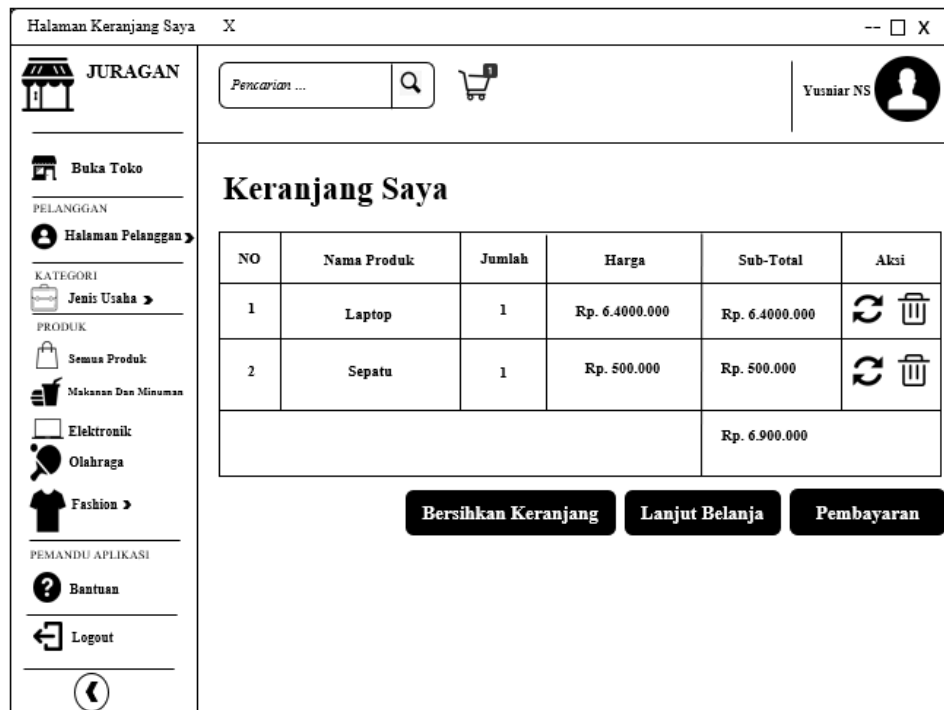
At the bottom of the form, there are two buttons: "Kembali" (Back) and "Simpan Produk" (Save Product).

*Gambar 5.51 UI Form Input Produk*

Pada gambar 5.51 merupakan UI dari *form input* produk. Dimana hanya *user* dengan level pelanggan yang dapat mengakses UI tersebut. UI tersebut dapat dibuka dengan cara mengklik pada *button* yang berada dalam gambar 5.50. *form* tersebut berfungsi untuk menambahkan suatu produk baru yang akan

di jual oleh pelanggan. Pelanggan diperlukan mengisi informasi – informasi yang dibutuhkan oleh sistem. Informasi tersebut akan di tampilkan dengan *textbox*. Terdapat dua *button* yang dimana berfungsi untuk memulai aksi *input* produk.

### 13). *User Interface* Halaman Keranjang



*Gambar 5.52 UI Halaman Keranjang*

Pada gambar 5.52 merupakan UI dari halaman keranjang yang dimana hanya dapat di akses oleh *user* dengan level pelanggan. Pada bagian *sidebar* dan *topbar* terlihat sama dengan gambar 5.45. Pada bagian *index* nya terdapat tabel yang isinya di peroleh dari hasil menambahkan produk kedalam keranjang melalui *button*. Terdapat 2 *icon* di dalam tabel tersebut yaitu *refresh* dan *delete* yang sudah di fungsikan. Di luar tabel terdapat 3 *button* yang dimana untuk memulai aksi yang diberikan.

## 5.4 Perancangan Arsitektur Perangkat Lunak Dan Perangkat Keras Sistem

Pada perancangan perangkat lunak dan perangkat keras tersebut merupakan kebutuhan – kebutuhan untuk mengakses aplikasi JURAGAN tersebut yang dimana telah dicantumkan pada tabel 5.23 – 5.24.

### A. Aplikasi Kebutuhan Perangkat Lunak

Perangkat lunak (*software*) yang dibutuhkan dalam aplikasi JURAGAN tersebut adalah sebagai berikut :

*Tabel 5.23 Software Requirement*

No.	Tool/Software	Description
1	Windows 7/8/10	Operation System
2	Xampp, VS Code, CodeIgniter, Microsoft Visio, Bizagi Modeler, Start UML, Microsoft Word	Development Tools
3	Browser	Web Browser Application
4	MariaDB	Database

### B. Aplikasi Kebutuhan Perangkat Keras

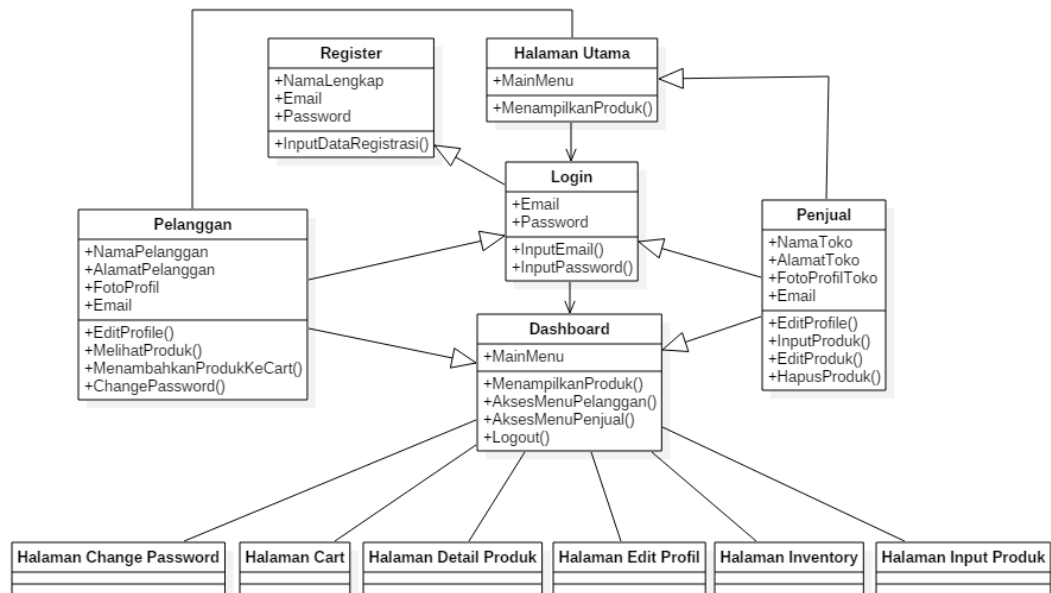
Perangkat keras (*hardware*) yang dibutuhkan dalam aplikasi JURAGAN tersebut adalah sebagai berikut :

*Tabel 5.24 Hardware Requirement*

No.	Divice Name	Specification
1	Processor	Core Inside (min)
2	Memory	DDR3 2 GB (min)
3	Harddisk	500 GB HD (min)
4	VGA	512 MB (min)

## 5.5 Pemetaan Struktur Diagram User/Aktor Sistem

Adapun pemetaan struktur diagram *user/aktor* pada sistem aplikasi JURAGAN tersebut adalah sebagai berikut :



Gambar 5.53 Pemetaan Struktur Diagram User/Aktor Pada Sistem JURAGAN