

**APLIKASI PEMESANAN MENU  
MAKANAN DAN MINUMAN  
BERBASIS *WEBSITE*  
MENGUNAKAN *FRAMEWORK*  
*CODEIGNITER* (*STUDI*  
*KASUS:RUPARUPI HANDICRAFT*  
*MARKET*)**



---

# **APLIKASI PEMESANAN MENU MAKANAN DAN MINUMAN BERBASIS *WEBSITE* MENGGUNAKAN *FRAMEWORK* *CODEIGNITER (STUDI* *KASUS:RUPARUPI HANDICRAFT* *MARKET)***

---

**Teddy Gideon Manik, Sr Rahayu**  
D4 Teknik Informatika



**Kreatif Industri Nusantara**

***Penulis:***

Rolly Maulana Awangga

ISBN : 978-602-53897-0-2

***Editor:***

M. Yusril Helmi Setyawan

***Penyunting:***

Syafrial Fachrie Pane

Khaera Tunnisa

Diana Asri Wijayanti

***Desain sampul dan Tata letak:***

Deza Martha Akbar

***Penerbit:***

Kreatif Industri Nusantara

***Redaksi:***

Jl. Ligar Nyawang No. 2

Bandung 40191

Tel. 022 2045-8529

Email : awangga@kreatif.co.id

***Distributor:***

Informatics Research Center

Jl. Sariasih No. 54

Bandung 40151

Email : irc@poltekpos.ac.id

Cetakan Pertama, 2019

Hak cipta dilindungi undang-undang

Dilarang memperbanyak karya tulis ini dalam bentuk dan dengan cara  
apapun tanpa ijin tertulis dari penerbit

*‘Jika Kamu tidak dapat  
menahan lelahnya  
belajar, Maka kamu harus  
sanggup menahan  
perihnya Kebodohan.’  
Imam Syafi’i*

# CONTRIBUTORS

---

ROLLY MAULANA AWANGGA, Informatics Research Center., Politeknik Pos Indonesia, Bandung, Indonesia





# CONTENTS IN BRIEF

---

<b>1</b>	<b>Bab 1 Pengantar</b>	<b>1</b>
<b>2</b>	<b>Analisis Dan Perancangan</b>	<b>3</b>
<b>3</b>	<b>Cara Instalasi Tools</b>	<b>29</b>



# DAFTAR ISI

---

Daftar Isi	iv
Daftar Gambar	ix
Daftar Tabel	xiii
Daftar Simbol	xiv
Daftar Lampiran	xvii
Foreword	xvii
Kata Pengantar	xix
Acknowledgments	xxi
Acronyms	xxiii
Glossary	xxv
List of Symbols	xxvii
Introduction	xxix
<i>Teddy Gideon Manik, Sri Rahayu</i>	

<b>1</b>	<b>Pengantar</b>	<b>1</b>
<b>2</b>	<i>Python</i>	<b>3</b>

2.1	Sejarah <i>Python</i>	3
2.2	Perbedaan <i>Python 2.x</i> dan <i>Python 3.x</i>	4
		<b>ix</b>
<b>x</b>	DAFTAR ISI	
2.3	Instalasi Python	11
2.3.1	Windows (Windows 10)	11
2.4	Instalasi Pip	16
2.4.1	Windows (Windows 10)	16
2.4.2	Linux (Ubuntu 19.04)	17
2.5	Setting Environment	18
2.5.1	Windows (Windows 10)	18
2.5.2	Linux (Ubuntu 19.04)	20
2.6	Command Line Interface/Interpreter	21
2.6.1	Windows (Windows 10)	21
2.6.2	Linux (Ubuntu 19.04)	22
<b>3</b>	<b>Judul Bagian Kedua</b>	<b>23</b>
3.1	Variabel	23
3.2	Input dan Output	24
3.3	Operasi Aritmatika	24
3.4	Perulangan	24
3.4.1	For	25
3.4.2	While	25
3.5	Kondisi	25
3.6	Error	27
3.7	Try Except	28
<b>4</b>	<b>Fungsi dan Kelas</b>	<b>29</b>
4.1	Teori	29
4.1.1	Fungsi	29
4.2	Package	30
4.3	Class, Object, Atribute, and Method	30
4.4	Pemanggilan Class	31
4.5	Pemakaian Package Fungsi Apabila File Didalam Folder	31
4.6	Pemakaian Package Kelas Apabila File didalam Folder	31
	Daftar Pustaka	33

# DAFTAR GAMBAR

---

2.1	Metode <i>Waterfall</i>	4
2.2	Flowmap yang Sedang Berjalan Pada Pemesanan Makanan dan Minuman	5
2.3	Flowmap Prosedur Login Pada Pelanggan, Admin, Staf dan Chef	6
2.4	Flowmap Prosedur Pemesanan Menu Makanan dan Minuman yang akan Dibangun	6
2.5	Use Case Diagram	7
2.6	Class Diagram	7
2.7	<i>Sequence Diagram Login Admin (UC1)</i>	8
2.12	<i>Sequence Diagram Login Chef (UC2)</i>	10
2.12	<i>Sequence Diagram Login Staf (UC3)</i>	10
2.12	<i>Sequence Diagram Login Pelanggan (UC4)</i>	10
2.12	<i>Sequence Diagram Kelola Data Makanan dan Minuman (UC5)</i>	10
2.12	<i>Sequence Diagram Kelola Data User (UC6)</i>	10
2.12	<i>Sequence Diagram Kelola Status Masakan (UC7)</i>	10

2.12	<i>Sequence Diagram</i> Kelola Status Pengantaran (UC8)	10
2.12	<i>Sequence Diagram</i> View Data Makanan dan Minuman (UC9)	10
2.12	<i>Sequence Diagram</i> Input Data Pesanan (UC10)	10
2.12	<i>Sequence Diagram</i> View Data Pesanan (UC11)	10
2.12	<i>Communication Diagram</i> Login Admin (UC1)	10
2.12	<i>Communication Diagram</i> Login Chef (UC2)	10
2.8	<i>Communication Diagram</i> Login Staf (UC3)	8
2.9	<i>Communication Diagram</i> Login Pelanggan (UC4)	9
2.10	<i>Communication Diagram</i> Kelola Data Makanan dan Minuman (UC5)	9
2.11	<i>Communication Diagram</i> Kelola Data User (UC6)	10
2.12	<i>Communication Diagram</i> Kelola Status Masakan (UC7)	10
2.12	<i>Communication Diagram</i> Kelola Status Pengantaran (UC8)	10
2.13	<i>Communication Diagram</i> View Data Makanan dan Minuman	11

2.14	<i>Communication Diagram Input Data Pesanan (UC10)</i>	12
2.15	<i>Communication Diagram View Data Pesanan (UC11)</i>	12
2.16	<i>Activity Diagram Login Admin (UC1)</i>	12
2.17	<i>Activity Diagram Login Chef (UC2)</i>	13
2.18	<i>Activity Diagram Login Staf (UC3)</i>	13
2.19	<i>Activity Diagram Login Pelanggan (UC4)</i>	14
2.20	<i>Activity Diagram Kelola Data Makanan dan Minuman (UC5)</i>	14
2.21	<i>Activity Diagram Kelola Data User (UC6)</i>	15
2.22	<i>Activity Diagram Kelola Status Masakan (UC7)</i>	15
2.23	<i>Activity Diagram Kelola Status Pengantaran (UC8)</i>	16
2.24	<i>Activity Diagram View Data Makanan dan Minuman UC9)</i>	16
2.25	<i>Activity Diagram Input Data Pesanan (UC10)</i>	17
2.26	<i>Activity Diagram View Data Pesanan (UC11)</i>	17
2.27	<i>Statechart Diagram Login Admin (UC1)</i>	18
2.28	<i>Statechart Diagram Login Chef (UC2)</i>	18
2.29	<i>Statechart Diagram Login Staf (UC3)</i>	19
2.30	<i>Statechart Diagram Login Pelanggan (UC4)</i>	19
2.31	<i>Statechart Diagram Kelola Data Makanan dan Minuman (UC5)</i>	20
2.32	<i>Statechart Diagram Kelola Data User (UC6)</i>	20
2.33	<i>Statechart Diagram Kelola Status Masakan (UC7)</i>	21
2.34	<i>Statechart Diagram Kelola Status Pengantaran (UC8)</i>	22
2.35	<i>Statechart Diagram Kelola Data User (UC9)</i>	22
2.36	<i>Statechart Diagram Kelola Status Masakan (UC10)</i>	22
2.37	<i>Statechart Diagram Kelola Status Pengantaran (UC11)</i>	22
2.38	<i>Component Diagram</i>	22
2.39	<i>Deployment Diagram</i>	22
2.40	<i>Halaman Form Login</i>	22
2.41	<i>Halaman Form Register</i>	22

2.42	<i>Halaman Home</i>	22
2.43	<i>Halaman Pemesanan Menu</i>	16
2.44	<i>Halaman Keranjang</i>	16
2.45	<i>Halaman Pembayaran</i>	16
2.46	<i>Halaman Admin</i>	16
2.47	<i>Halaman Admin Bagian Data Produk</i>	16
2.48	<i>Halaman Admin Bagian Data User</i>	16
2.49	<i>Halaman Admin Bagian Data Customer</i>	16
2.50	<i>Halaman Admin Bagian Pesanan</i>	16
2.51	<i>Halaman Admin Laporan</i>	16
2.52	<i>Halaman Chef</i>	16
2.53	<i>Chef Bagian Produk</i>	16
2.54	<i>Halaman Chef Bagian Customer</i>	16
2.55	<i>Halaman Chef Bagian Pesanan</i>	16
2.56	<i>Halaman Chef Bagian Laporan</i>	16
2.57	<i>Halaman Staf</i>	16
2.58	<i>Halaman Staf Bagian Produk</i>	16
2.59	<i>Halaman Staf Bagian Customer</i>	16
2.60	<i>Halaman Staf Bagian Pesanan</i>	16
2.61	<i>Halaman Staf Bagian Laporan</i>	16
2.62	<i>Struktur File pada bagian Model</i>	16
2.63	<i>Struktur File pada bagian Controller</i>	16
2.64	<i>Struktur File pada bagian View</i>	16
2.65	<i>Halaman Form Login</i>	16
2.66	<i>Halaman Form Register</i>	16
2.67	<i>Halaman Home</i>	16
2.68	<i>Halaman Pemesanan Menu</i>	
2.69	<i>Halaman Keranjang</i>	
2.70	<i>Halaman Pembayaran</i>	



2.71	Halaman Admin
2.72	Halaman Admin Bagian Data Produk
2.73	Halaman Admin Bagian Data <i>User</i>
2.74	Halaman Admin Bagian Data <i>Customer</i>
2.75	Halaman Admin Bagian Pesanan
2.76	Halaman Admin Laporan
2.77	Halaman Chef
2.78	Chef Bagian Produk
2.79	Halaman Chef Bagian <i>Customer</i>
2.80	Halaman Chef Bagian Pesanan
2.81	Halaman Chef Bagian Laporan
2.82	Halaman Staf
2.83	Halaman Staf Bagian Produk
2.84	Halaman Staf Bagian <i>Customer</i>
2.85	Halaman Staf Bagian Pesanan
2.86	Halaman Staf Bagian Laporan

# DAFTAR TABEL

---

2.14	Tinjauan Studi
2.15	Dokumen Pemesanan
2.16	Dokumen Pembayaran
2.17	Kebutuhan Perangkat Keras
2.18	Kebutuhan Perangkat Lunak
2.19	Definisi Aktor
2.20	Definisi <i>Use Case</i>
2.21	Skenario <i>Use Case Login</i> Admin
2.22	Skenario <i>Use Case Login</i> Chef
2.23	Skenario <i>Use Case Login</i> Staf
2.24	Skenario <i>Use Case Login</i> Pelanggan
2.25	Skenario <i>Use Case</i> Kelola Data Makanan dan Minuman
2.26	Skenario <i>Use Case</i> Kelola Data <i>User</i>

- 2.27      Skenario *Use Case* Kelola Status Masakan
- 2.28      Skenario *Use Case* Kelola Status Pengantaran
- 2.29      Skenario *Use Case View* Data Makanan dan Minuman
- 2.30      Skenario *Use Case Input* Data Pesanan
- 2.31      Skenario *Use Case View* Data Pesanan
- 2.32      Deskripsi Perangkat Keras
- 2.33      Deskripsi Perangkat Lunak
- 2.34      Identifikasi Pengujian



# Listings

---



# FOREWORD

---

Sepatah kata dari Kaprodi, Kabag Kemahasiswaan dan Mahasiswa





# KATA PENGANTAR

---

Segala Puji syukur kami senantiasa ucapkan atas kehadiran Allah SWT karena curahan rahmat serta karunia-nya lah kami akhirnya sampai pada tahap menyelesaikan laporan dengan judul Aplikasi Pemesanan Menu Makanan dan Minuman Berbasis *Website* Menggunakan *Framework* Codeigniter (Studi Kasus: Family Brasil S3). Ucapan terimakasih kami senantiasa ucapkan kepada orang tua kami yang telah memberikan motivasi serta nasihat kepada kami, karena berkat doa serta dukungannya kami dapat terus melanjutkan pendidikan di Politeknik Pos Indonesia. Kami juga ucapkan rasa terimakasih yang sebanyak-banyaknya untuk : Noviana Riza, S.Si., M.T. Selaku dosen pembimbing yang telah membimbing kami untuk dapat menyelesaikan laporan project 3 dengan tepat waktu. Kami sungguh-sungguh berharap laporan ini bisa bermanfaat kepada masyarakat dengan tujuan untuk meningkatkan pengetahuan sekaligus wawasan terkait pengajuan proposal karya ilmiah, sekaligus mengetahui permasalahan dan cara untuk mengatasinya. Kami sadar bahwa laporan ini masih perlu diperbaiki kembali. Dengan demikian, kami benar-benar menantinya adanya kritik dan saran untuk perbaikan laporan yang hendak kami tulis dimasa selanjutnya, menyadari tidak ada suatu hal yang sempurna tanpa disertai saran yang konstruktif. Kami berharap laporan sederhana ini bisa dimengerti oleh setiap pihak terutama untuk para pembaca. Kami mohon maaf yang sebesar-besarnya jika ada perkataan yang kurang berkenan di hati.

T. G. Manik., Sri Rahayu



# ACKNOWLEDGMENTS

---

Terima kasih atas semua masukan dari para mahasiswa agar bisa membuat buku ini lebih baik dan lebih mudah dimengerti.

Terima kasih ini juga ditujukan khusus untuk team IRC yang telah fokus untuk belajar dan memahami bagaimana buku ini mendampingi proses Intership.

R. M. A.



# ACRONYMS

---

ACGIH	American Conference of Governmental Industrial Hygienists
AEC	Atomic Energy Commission
OSHA	Occupational Health and Safety Commission
SAMA	Scientific Apparatus Makers Association



# GLOSSARY

---

git	Merupakan manajemen sumber kode yang dibuat oleh linus torvald.
bash	Merupakan bahasa sistem operasi berbasiskan *NIX.
linux	Sistem operasi berbasis sumber kode terbuka yang dibuat oleh Linus Torvald





# SYMBOLS

---

- $A$  Amplitude
- $\&$  Propositional logic symbol
- $a$  Filter Coefficient
  
- $B$  Number of Beats



# PENGANTAR

---

Segala Puji Syukur bagi Allah SWT yang telah menganugerahkan rahmat dan hidayah-Nya kepada kita semua. Salawat serta salam semoga selalu terlimpahkan pada Rasulullah SAW, kepada keluarganya, para sahabatnya, dan semoga sampai pada ummatnya hingga akhir zaman.

Alhamdulillahirabbil'alamin, laporan dengan judul **“Aplikasi Pemesanan Menu Makanan dan Minuman Berbasis Website Menggunakan Framework Codeigniter (Studi Kasus: Family Brasil S3)”** dalam memenuhi kelulusan mata kuliah Proyek TI III pada Semester V Tahun Akademik 2019/2020 ini, dapat kami tuntaskan. Kami ucapkan terima kasih kepada semua pihak yang telah mendukung kami, terkhusus kami sampaikan kepada:

1. M. Yusril Helmi Setyawan, S.Kom., M.Kom. selaku Ketua Program Studi D IV Teknik Informatika.
2. Cahyo Prianto, S.Pd., M.T.selaku Koordinator Proyek TI III.
3. M. Harry K Saputra, S.T., M.T.I selaku Dosen Wali D4 TI 3B dan Dosen Pembimbing yang telah memberikan banyak pengarahan dan masukan dalam pembuatan Proyek TI III ini.
4. Rekan-rekan Prodi TI Kelas 3B, yang selalu semangat menyongsong masa depan yang cemerlang.
5. Serta semua pihak yang telah memberikan dukungan dan bantuan dalam proses pembuatan Proyek TI III ini.

Dalam penyusunan laporan ini, masih terdapat kesalahan. Untuk itu, mohon maaf atas kekurangan dalam penulisan dalam laporan ini. Besar harapan kami, laporan ini dapat digunakan sebagaimana mestinya dan dapat bermanfaat bagi seluruh pihak yang terkait dalam kegiatan perkuliahan.

# BAB 1 PENGANTAR

---

## 1.1 Aplikasi

Aplikasi adalah kumpulan dari perintah program yang di buat dengan tujuan melakukan pekerjaan-pekerjaan tertentu (khusus)[1].

## 1.2 Pemesanan

Pemesanan adalah suatu aktifitas yang dilakukan oleh konsumen sebelum membeli. Untuk mewujudkan kepuasan konsumen maka perusahaan harus mempunyai sebuah sistem pemesanan yang baik[2].

## 1.3 Website

*Website* adalah kumpulan halaman *web* yang memiliki hubungan satu sama lain dan isi filenya saling terkait. *Web* terdiri dari halaman, dan kumpulan halaman yang dinamakan *homepage*. *Homepage* mempunyai posisi yang sangat tertinggi dari halaman-halaman yang terkait di bawahnya[3].

## 1.4 Web Service

*Web service* merupakan suatu penghubung antara *server* ke *client* dengan menggunakan jaringan. Di bagian klien tidak hanya dapat berupa *web*, aplikasi juga bisa menerima atau berbagi informasi ke bagian *server*. Dengan menggunakan *web service* dapat memberikan pelayanan yang dibutuhkan oleh permintaan klien. Klien dari *web service* tidak hanya berupa aplikasi *web*, tetapi juga bisa sebuah aplikasi *enterprise*. Tujuan utama dari teknologi *web service* adalah untuk memungkinkan aplikasi pada *platform* yang berbeda

untuk bertukar data bisnis [4].

## ***1.5 Unified Modeling Language (UML)***

UML merupakan bahasa penulisan yang direpresentasikan dalam bentuk notasi atau diagram-diagram yang berbeda yang dapat digunakan untuk memodelkan sistem perangkat lunak. UML juga memberikan standar penulisan yang tidak bergantung pada bahasa pemrograman tertentu, sebagai *blue print* yang meliputi konsep proses bisnis, penulisan kelas-kelas, skema *database* dan komponen-komponen yang diperlukan dalam sistem *software*[5]. UML mendefinisikan diagram-diagram sebagai berikut:

### ***1. Use Case Diagram***

*Use Case Diagram* merupakan diagram utama yang sangat penting untuk mengorganisasi dan memodelkan suatu sistem yang dibutuhkan serta diharapkan pengguna [6].

### ***2. Activity Diagram***

*Activity Diagram* menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir [6].

### ***3. Class Diagram***

*Class Diagram* adalah sebuah spesifikasi yang jika diinstantiasikan akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek[6].

### ***4. Sequence Diagram***

*Sequence diagram* adalah suatu diagram yang menggambarkan

interaksi antar objek dan mengindikasikan komunikasi diantara objek-objek di dalam dan disekitar sistem [7].

## 5. *Component Diagram*

Hal penting pada *component* adalah *component* mewakili potongan-potongan yang independen yang bisa dipesan dan diperbaharui sewaktu-waktu [8].

## 6. *Deployment Diagram*

*Deployment diagram* menunjukkan tata letak sebuah sistem secara fisik, menampakkan bagian-bagian *software* yang berjalan pada bagian *hardware*. Bagian utama *hardware* atau perangkat keras adalah node yaitu nama umum untuk semua jenis sumber komputasi [8].

### 1.6 *Hypertext Preprocessor (PHP)*

PHP merupakan singkatan dari *Hypertext Preprocessor* yaitu Bahasa pemrograman berbasis *web*. PHP adalah bahasa pemrograman yang digunakan untuk membuat aplikasi berbasis *web* [9].

PHP adalah *script* yang digunakan untuk membuat halaman *web* yang bersifat dinamis. Dinamis berarti halaman yang akan ditampilkan dibuat suatu halaman itu diminta oleh *client*. PHP juga bersifat *open source* sehingga setiap orang dapat menggunakan secara gratis [8].

PHP juga merupakan *script* yang terintegrasi dengan HTML. Selain itu, PHP bersifat *server side*, yang artinya hanya dapat berjalan pada sisi *server* saja dan tidak dapat berfungsi tanpa adanya sebuah *server* di dalamnya. PHP juga sudah mendukung atau dapat dijalankan



pada berbagai sistem operasi [10].

## 1.7 CodeIgniter

*CodeIgniter* adalah sebuah *framework* yang digunakan untuk membuat sebuah aplikasi berbasis *web* yang disusun dengan menggunakan bahasa PHP. Di dalam *CodeIgniter* terdapat beberapa macam kelas (*class*) yang berbentuk *library* dan *helper* [11].

Berikut ini adalah cara kerja dari *CodeIgniter* [12]:

- a. *index.php* berlaku sebagai *controller* di depan, yang menginisiasikan sumber daya yang dibutuhkan untuk menjalankan *codeigniter*
- b. *Router* akan memeriksa *request HTTP* untuk menentukan apa yang harus dilakukan dengan *request* tersebut,
- c. Jika ada *file cache*, maka *file cache* langsung dikirimkan kepada *browser*, dengan memotong eksekusi sistem normal,
- d. Untuk keamanan, sebelum suatu aplikasi *controller* dipanggil, *request HTTP* dan data yang dikirimkan oleh pengguna disaring (*filter*) terlebih dahulu, *Controller* akan memuat *model*, *library*, *helper*, dan sumber daya lainnya yang diperlukan untuk memproses *request* tersebut.
- e. *View* terakhir di-*render* kemudian dikirimkan kepada *browser* untuk dilihat. Jika *caching* didefinisikan, maka *view* akan di-*cache* terlebih dahulu, setelah itu baru dikirimkan kepada *browser*.

*CodeIgniter* menerapkan konsep *Model*, *View* dan *Controller* (MVC), di mana MVC tersebut akan sebuah aplikasi membagi menjadi 3

kelompok, yaitu:

- a. *Model* merupakan struktur data. Secara spesifik *class model* akan mengandung fungsi kode yang akan membantu dalam segala proses yang berhubungan dengan *database* seperti memasukkan, mengedit mendapatkan dan menghapus data dalam sebuah *database* [13].
- b. *View* adalah komponen yang menampilkan antarmuka (*user interface*) untuk pengguna aplikasi. Antarmuka ini dibuat berdasarkan data dari *model* [12].
- c. *Controller* berfungsi sebagai *regulator* interaksi antara *model* dan *view*. *Controller* juga berperan untuk menerima *request* dan data dari *user* kemudian menentukan apa yang harus diproses oleh aplikasi [5].

## 1.8 Database

*Database* adalah kumpulan dari data-data yang membentuk suatu berkas (*file*) yang saling berhubungan (*relation*) yang diorganisasi sedemikian rupa dengan tata cara untuk membentuk data baru atau informasi sehingga kelak dapat dimanfaatkan dengan cepat dan mudah [9].

*Database* pada awalnya relatif "*flat*", artinya mereka hanya terbatas pada baris dan kolom yang sederhana, yang hanya dapat menyimpan teks atau data numerik, seperti *spreadsheet*. Namun, dalam perkembangannya, saat ini *database* memungkinkan pengguna menyimpan tipe data lain seperti klip suara, gambar, dan video [14].

## **1.9 MariaDB**

MariaDB adalah sistem manajemen *database* relasional yang dikembangkan dari MySQL. MariaDB dikembangkan komunitas pengembang yang sebelumnya berkontribusi untuk *database* MySQL [15]. MariaDB adalah DBMS hasil *forking* dari DBMSMySQL. Jadi, *syntax query* yang digunakan hampir sama [16].

## **1.10 Metode Waterfall**

Metode *Waterfall* merupakan model pengembangan sistem informasi yang sistematis dan sekuensial. Metode *waterfall* memiliki tahapan-tahapan sebagai berikut [17]:

### **1. *Requirements analysis and definition***

Layanan sistem, kendala, dan tujuan ditetapkan oleh hasil konsultasi dengan pengguna yang kemudian didefinisikan secara rinci dan berfungsi sebagai spesifikasi sistem.

### **2. *System and software design***

Tahapan perancangan sistem mengalokasikan kebutuhan-kebutuhan sistem baik perangkat keras maupun perangkat lunak dengan membentuk arsitektur sistem secara keseluruhan. Perancangan perangkat lunak melibatkan identifikasi dan penggambaran abstraksi sistem dasar perangkat lunak dan hubungannya.

### **3. *Implementation and unit testing***

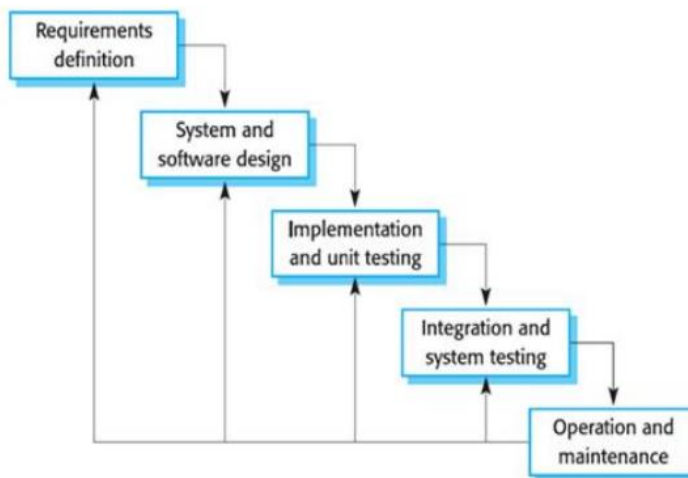
Pada tahap ini, perancangan perangkat lunak direalisasikan sebagai serangkaian program atau unit program. Pengujian melibatkan verifikasi bahwa setiap unit memenuhi spesifikasinya.

#### 4. *Integration and system testing*

Unit-unit individu program atau program digabung dan diuji sebagai sebuah sistem lengkap untuk memastikan apakah sesuai dengan kebutuhan perangkat lunak atau tidak. Setelah pengujian, perangkat lunak dapat dikirimkan ke *customer*.

#### 5. *Operation and maintenance*

Biasanya (walaupun tidak selalu), tahapan ini merupakan tahapan yang paling panjang. Sistem dipasang dan digunakan secara nyata. *Maintenance* melibatkan pembetulan kesalahan yang tidak ditemukan pada tahapan-tahapan sebelumnya, meningkatkan implementasi dari unit sistem, dan meningkatkan layanan sistem sebagai kebutuhan baru.





## BAB 2

---

# ANALISIS DAN PERANCANGAN

---

### 2.1 Analisis Sistem

Analisis adalah tahap yang sangat penting karena suatu kesalahan dalam tahap ini akan mempengaruhi pada tahap berikutnya. Penelitian juga membuktikan bahwa kesalahan yang diperbaiki setelah tahap analisis akan memakan biaya yang lebih besar dari pada jika di perbaiki saat dilakukan analisis.

Analisis system merupakan penguraian dari suatu system yang utuh ke dalam bagian-bagian komponennya dengan maksud untuk mengidentifikasikan dan mengevaluasi permasalahan-permasalahan, kesempatan, hambatan yang terjadi dan kebutuhan yang di harapkan sehingga dapat di usulkan perbaikan-perbaikannya. Hal-hal yang akan dianalisis pada tahap analisis *system* ini adalah analisis prosedur system yang sedang berjalan, analisis aliran informasi, analisis pengkodean, analisis basis data analisis kebutuhan nonfungsional.

## **2.1.1 Analisis Sistem yang Sedang Berjalan**

Sistem pemesanan yang sedang berjalan saat ini masih dilakukan di *counter* pesan, kemudian pesanan dicatat oleh kasir dan diserahkan ke staf dapur.

### **2.1.1.1. Analisis Prosedur**

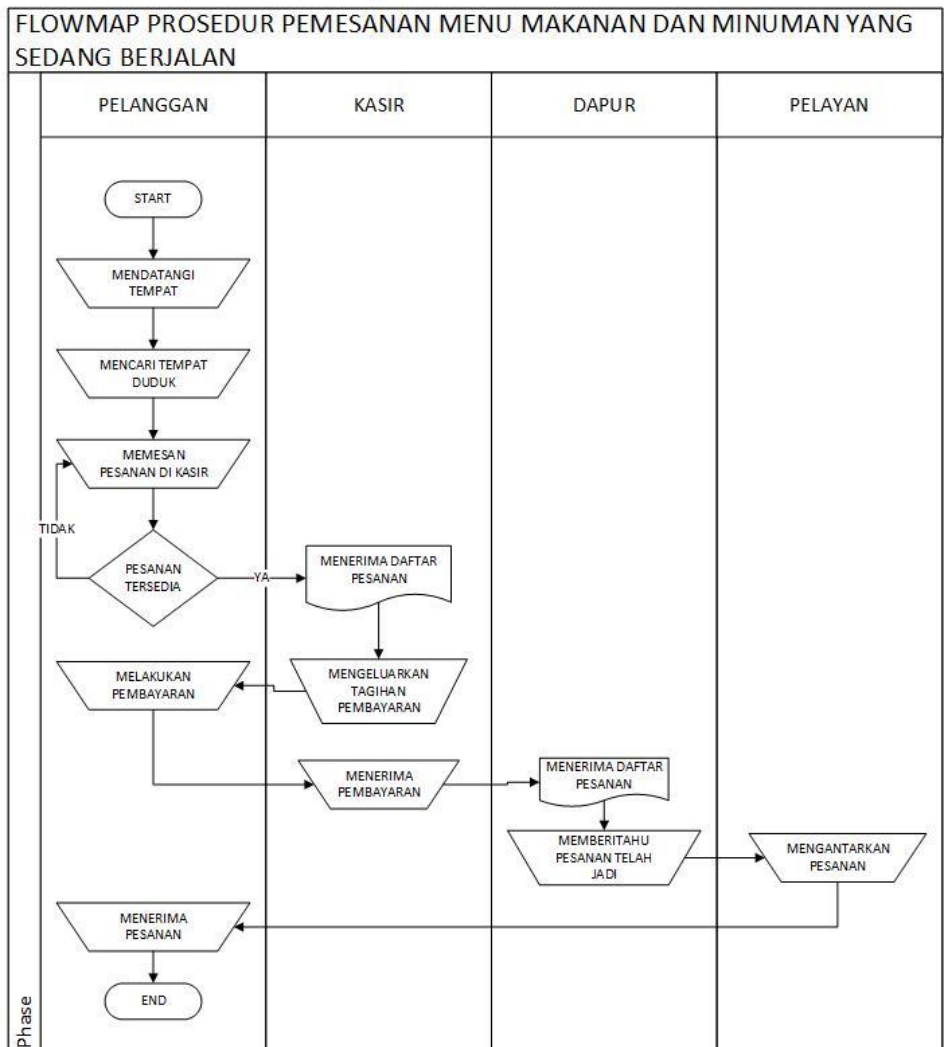
#### **A. Analisis Prosedur yang Sedang Berjalan**

Pemesanan makanan adalah kegiatan yang dilakukan pelanggan, untuk melakukan pemesanan makanan dan minuman kepada pihak *family brasil*. Prosedur yang ada pada pemesanan makanan dan minuman yang sedang berjalan, yaitu sebagai berikut:

1. Pelanggan mendatangi *family brasil* dan mencari tempat duduk.
2. Pelanggan mendatangi meja pesan untuk memesan makanan dan minuman sekaligus membayarnya.
3. Kasir menyampaikan pesanan ke bagian staf dapur.
4. Setelah catatan pesanan sampai ke staf dapur, selanjutnya staf dapur akan menyiapkan makanan dan minuman yang dipesan.
5. Apabila makanan dan minuman yang disiapkan telah selesai, maka staf dapur akan langsung mengantarkan makanan dan minuman ke meja pelanggan.
6. Pelanggan menerima pesanan.



Adapun *flowmap* prosedur pemesanan makanan dan minuman yang sedang berjalan, yaitu:



Gambar 2. 1 Flowmap yang Sedang Berjalan Pada Prosedur Pemesanan Makanan dan Minuman

### 2.1.1.2 Analisis Dokumen yang Digunakan

Dari hasil analisis yang dilakukan, dokumen yang digunakan di antaranya adalah dokumen

pemesanan dan dokumen pembayaran.

*Tabel 2.1 Dokumen Pemesanan*

<b>Dibuat oleh</b>	Kasir
<b>Dibuat untuk</b>	Staf Dapur
<b>Isi</b>	Berupa daftar pesanan
<b>Frekuensi</b>	Dibuat sesuai menu yang disediakan
<b>Tujuan</b>	Menyediakan menu yang akan dipesan oleh penumpang

*Tabel 2.2 Dokumen Pembayaran*

<b>Dibuat oleh</b>	Kasir
<b>Dibuat untuk</b>	<i>Family Brasil S3</i>
<b>Isi</b>	Berupa daftar tagihan pembayaran
<b>Frekuensi</b>	Dibuat sesuai menu yg dipesan
<b>Tujuan</b>	Menyediakan laporan rekap pemasukan ke <i>coffee shop</i>

### **2.1.2 Analisis Sistem yang akan Dibangun**

#### **A. Analisis Sistem yang akan Dibangun pada Prosedur**

##### ***Login Pelanggan, Admin, Chef***

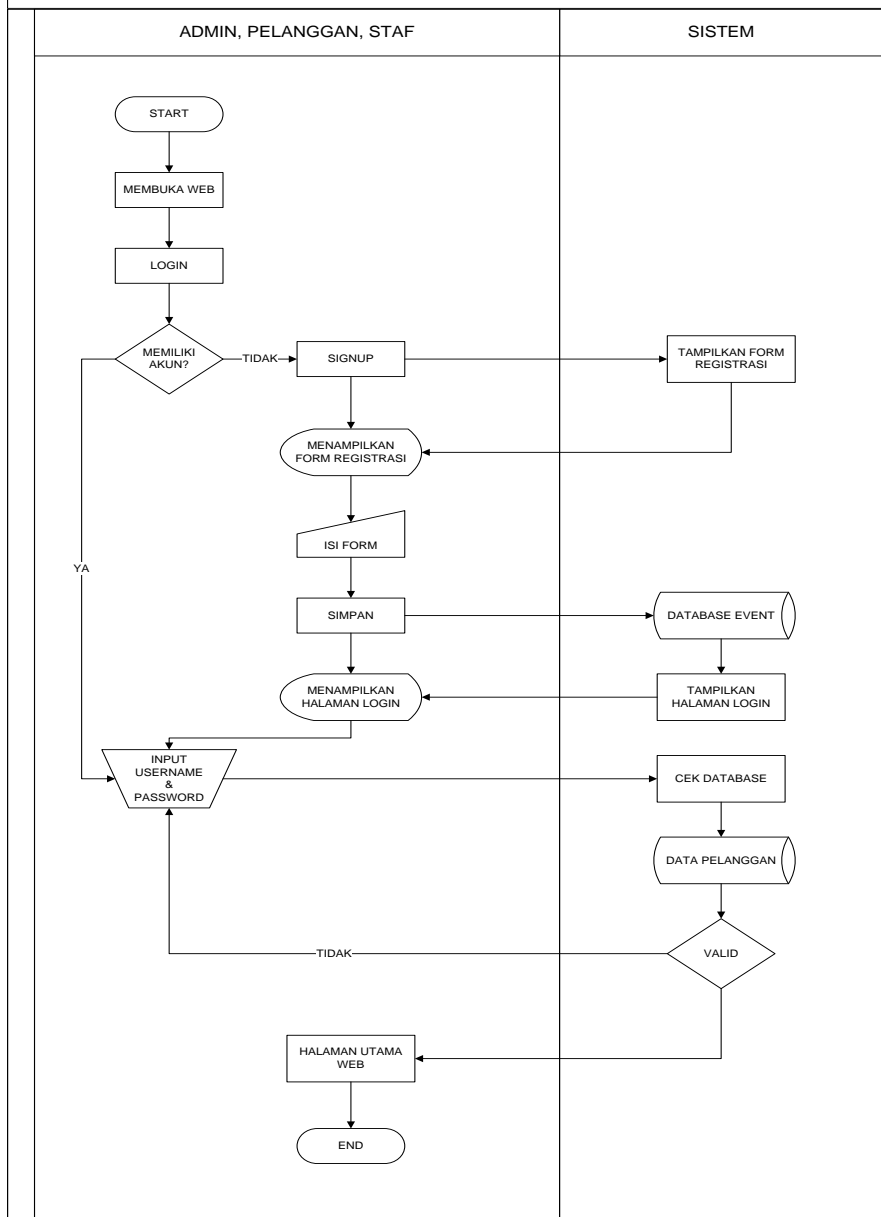
Pada prosedur login admin ini melibatkan 4 entitas (aktor) yaitu Admin, *Chef*, Staf, dan Pelanggan. Di mana setiap entitas (aktor) tersebut memiliki hak akses yang berbeda terhadap sistem atau aplikasi yang dibuat. Prosedur yang sudah dibangun pada *login*, yaitu:

1. Pelanggan, Admin, dan Staf harus login untuk bisa masuk ke halaman utama web.

2. Pelanggan, Admin, dan Staf harus menginputkan *username* dan *password* untuk bisa masuk ke halaman utama *web*.
3. Sistem akan mengecek *username* dan *password*. Apabila *username* dan *password* sudah *valid* dengan data yang sudah tersimpan di *database*, maka akan langsung masuk ke halaman utama *web*.
4. Apabila belum terdaftar maka melakukan registrasi.

Adapun *flowmap* prosedur *login* Pelanggan, Admin, *Chef*, dan Staf adalah sebagai berikut:

# FLOWMAP LOGIN



*Gambar 3. 2 Flowmap Prosedur Login pada Pelanggan, Admin, Chef, dan Staf*

## **B. Analisis Sistem yang akan Dibangun pada Prosedur**

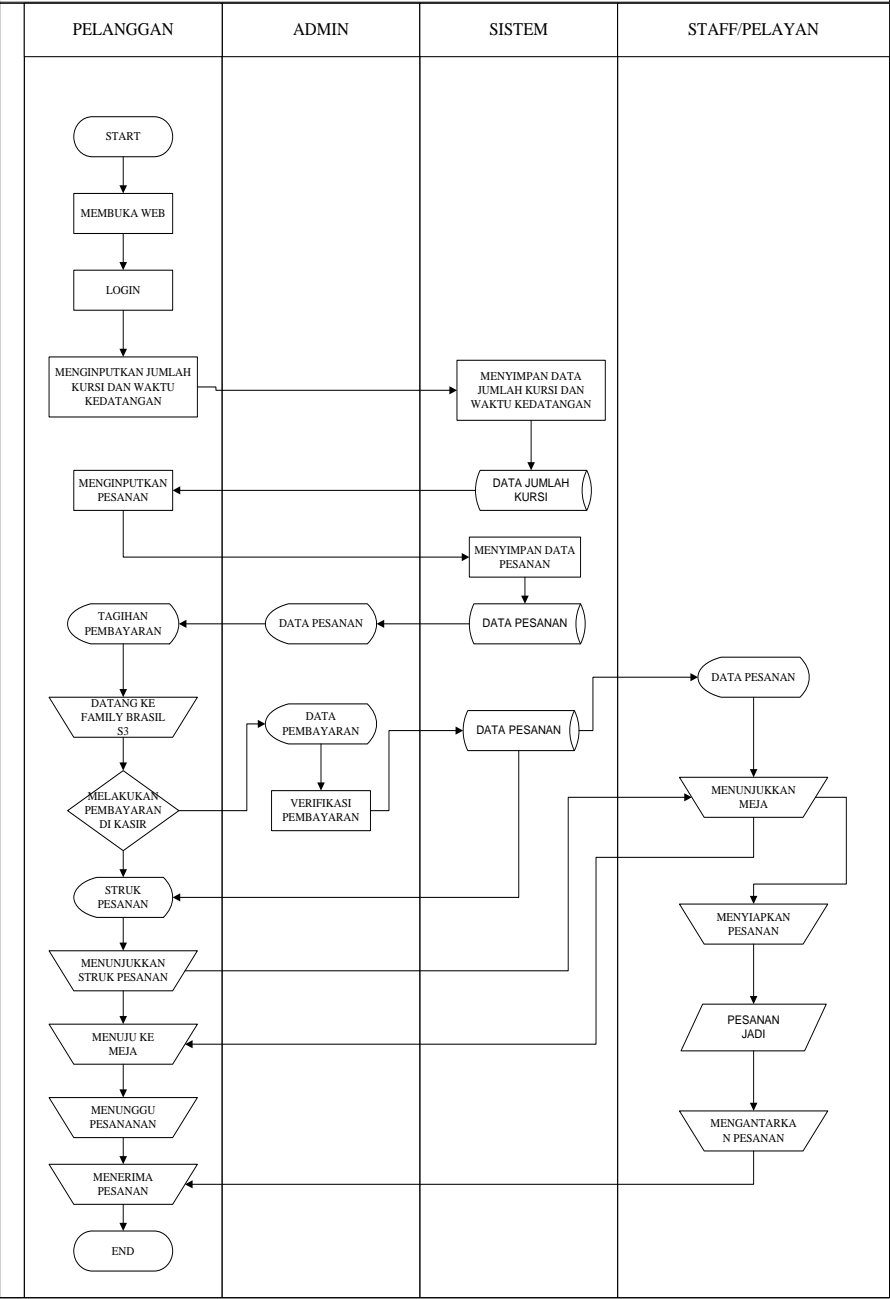
### **Pemesanan Menu Makanan dan Minuman**

Pada prosedur pemesanan ini melibatkan 4 (entitas) aktor yaitu Pelanggan, Admin, *Chef*, dan Staf. Prosedur pada pemesanan makanan, yaitu:

1. Pelanggan membuka *web* untuk menginputkan jumlah kursi dan estimasi waktu kedatangan, kemudian memesan makanan dan minuman.
2. Sistem akan menyimpan jumlah kursi, estimasi waktu kedatangan dan data pesanan makanan dan minuman.
3. Pelanggan menerima tagihan pembayaran dan mendatangi tempat FAMILY BRASIL S3. Jika sudah melakukan pembayaran maka akan diverifikasi oleh Admin dan data pembayaran akan tersimpan di sistem
4. Pelanggan menerima struk pesanan untuk ditunjukkan ke staff
5. Kemudian Staf menunjukkan kursi yang tersedia untuk pelanggan tersebut.
6. *Chef* menyiapkan pesanan pelanggan berdasarkan data pesanan yang diterima, jika pesanan sudah jadi maka *Chef* akan mengubah status pesanan bahwa pesanan sudah jadi.
7. Staf mengantarkan pesanan yang sudah jadi ke pelanggan dan Staf akan mengubah status pesanan bahwa pesanan sedang diantar.
8. Pelanggan menerima makanan dan minuman yang dipesan sebelumnya.

Adapun *flowmap* prosedur pemesanan menu makanan dan minuman yang akan dibangun adalah sebagai berikut:

FLOWMAP PROSEDUR PEMESANAN MAKANAN DAN MINUMAN YANG AKAN DIBANGUN



### *Gambar 3. 3 Flowmap Prosedur Pemesanan Menu Makanan dan Minuman yang akan Dibangun*

#### 3.1.2.1 Kebutuhan Fungsional (*Functional Requirements*)

Analisis kebutuhan fungsional merupakan suatu kebutuhan yang berhubungan dengan kebutuhan sistem yang akan dibuat. Di mana menjabarkan mengenai fungsi-fungsi yang dapat mendukung jalannya sistem.

Setiap proses memiliki representasi masing-masing pada sebuah tabel atau data yang terdapat pada *database* yang telah dirancang sebelumnya dan setiap proses berhubungan langsung dengan entitas atau *user*.

#### 3.1.2.2 Kebutuhan Non-Fungsional (*Non-Functional Requirements*)

Analisis kebutuhan non fungsional dilakukan untuk mengetahui spesifikasi kebutuhan untuk sistem. Spesifikasi kebutuhan melibatkan analisis perangkat keras, analisis perangkat lunak, dan analisis pengguna.

#### **A. Kebutuhan Perangkat Keras Saat Pembuatan Aplikasi**

*Tabel 3.3 Kebutuhan Perangkat Keras*

No	Nama Perangkat	Spesifikasi	Keterangan
1.	<i>Hardisk</i>	100 GB	Media untuk menyimpan data aplikasi yang dibuat
2.	<i>RAM</i>	8 GB	Mempercepat pemrosesan data pada PC
3.	<i>Processor</i>	<i>Intel ® Core™ i5-6200U @2.30GHz</i>	Untuk kecepatan transfer data dari sistem yang sangat bergantung pada



			kecepatan prosesor komputer
--	--	--	-----------------------------

## B. Kebutuhan Perangkat Lunak Saat Pembuatan Aplikasi

*Tabel 3.4 Kebutuhan Perangkat Lunak*

No	Tools / Software	Fungsi	Keterangan
1.	<i>Windows 10</i>	Sistem Operasi	Kompatibel di semua perangkat
2.	<i>XAMPP 5.6</i>	<i>Server</i> yang berdiri sendiri ( <i>localhost</i> )	Dapat mengakses <i>web</i> dinamis tanpa koneksi internet dan <i>hosting</i>
4.	<i>Microsoft Office Visio 2016</i>	<i>Software</i> Pendukung	<i>Tools</i> nya sangat mudah dipahami, dan cara penggunaannya yang mudah
5.	<i>PHP</i>	<i>Bahasa Pemrograman</i> yang digunakan	Bersifat <i>open source</i> dan dapat membuat <i>web</i> menjadi dinamis
6.	<i>Sublime Text</i>	<i>Editor teks</i>	Memiliki banyak <i>fiture</i> , <i>cross-platform</i> , mudah, dan sederhana, dan mendukung banyak Bahasa pemrograman dan <i>markup</i>
7.	<i>CodeIgniter</i>	<i>Framework</i>	Memberikan standar <i>coding</i> sehingga memudahkan untuk mempelajari kembali sistem aplikasi yang dibangun
8.	<i>Google Chrome</i>	<i>Browser</i>	Memiliki proses <i>loading</i> pada tab yang terpisah
9.	<i>Star UML</i>	<i>Desain UML</i>	Mudah dalam menghubungkan antar koneksi pada suatu kelas atau aktivitas.

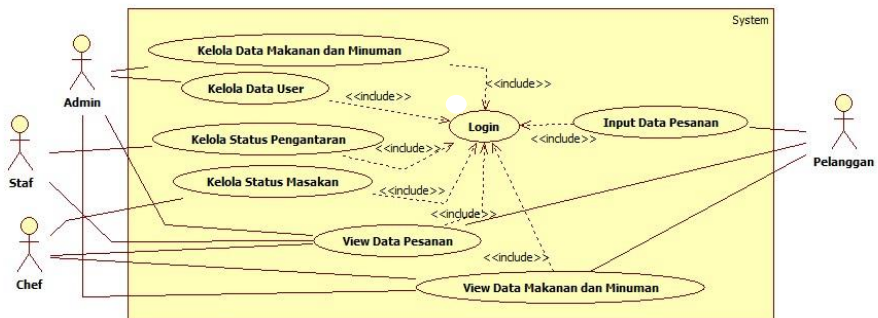
## A. Analisis Pengguna

Aplikasi yang akan dibuat ini digunakan untuk lingkungan masyarakat umum yang ingin berkunjung ke *Family Brasil S3*. Pengguna aplikasi yang terlibat adalah Admin, *Chef* dan Pelanggan

### 3.2 Perancangan

#### 3.2.1 Use Case Diagram

*Use Case Diagram* merupakan diagram utama yang sangat penting untuk mengorganisasi dan memodelkan suatu sistem yang dibutuhkan serta diharapkan pengguna [6].



Gambar 3. 4 Use Case Diagram

##### 3.2.1.1 Definisi Aktor

Tabel 3.5 Definisi Aktor

No	Aktor	Keterangan
1.	Admin	Melihat data pesanan, melakukan pengolahan data seperti kelola data makanan dan minuman, melihat menu makanan dan minuman dan kelola data <i>user</i> .
2.	<i>Chef</i>	Melihat data pesanan, melihat menu makanan dan minuman, dan melakukan pengelolaan data seperti kelola status masakan

		apakah masakan sudah selesai atau belum.
3.	Staf	Melihat data pesanan dan melakukan pengelolaan data seperti kelola status pengantaran pesanan apakah pesanan sudah diantar ke pelanggan atau belum.
4.	Pelanggan	Melihat menu makanan dan minuman, melakukan <i>input</i> data pesanan, dan melihat data pesanan.

### 3.2.1.2 Definisi *Use Case*

*Use case* yang ada dalam diagram didefinisikan pada tabel di bawah ini:

*Tabel 3.6 Definisi Use Case*

No .	No. Definisi	Use Case	Deskripsi
1.	UC1	<i>Login Admin</i>	Melakukan proses <i>login</i>
2.	UC2	<i>Login Chef</i>	Melakukan proses <i>login</i>
3.	UC3	<i>Login Staf</i>	Melakukan proses <i>login</i>
4.	UC4	<i>Login Pelanggan</i>	Melakukan proses <i>login</i>
5.	UC5	Kelola Data Makanan dan Minuman	Merupakan serangkaian aktivitas yang dilakukan untuk mengelola data menu makanan dan minuman.
6.	UC6	Kelola Data <i>User</i>	Merupakan serangkaian aktivitas yang dilakukan untuk mengelola data <i>user</i> .
7.	UC7	Kelola Status Masakan	Merupakan serangkaian aktifitas yang dilakukan untuk mengelola status masakan, apakah pesanan sudah selesai disiapkan atau belum.
8.	UC8	Kelola Status Pengantaran	Merupakan serangkaian aktifitas yang dilakukan untuk mengelola status pengantaran, apakah pesanan sudah diantar ke pelanggan atau belum.
9.	UC9	View Data Makanan dan Minuman	Merupakan aktifitas yang dilakukan untuk data menu makanan dan minuman yang tersedia.
10.	UC10	Input Data Pesanan	Merupakan aktifitas yang dilakukan untuk menginputkan menu makanan dan minuman yang dipesan..
11.	UC11	View Data Pesanan	Merupakan aktifitas yang dilakukan untuk melihat data pesanan.

### 3.2.1.3 Skenario *Use Case*

Skenario *Use Case* pada Aplikasi Pemesanan Menu Makanan dan Minuman Berbasis *Website* Menggunakan

Framework CodeIgniter (Studi Kasus: Family Brasil S3) di antaranya:

1. Skenario Use Case Login Admin

Skenario dalam diagram use case Login, ada pada tabel seperti berikut ini:

Tabel 3.7 Skenario Use Case Login Admin

Identifikasi	
Nomor	UC1
Nama	Login Admin
Tujuan	Melakukan Login untuk masuk halaman utama.
Deskripsi	Login dengan memasukan Username dan Password
Aktor	Admin
Skenario Utama	
Kondisi Awal	Menampilkan Form Login.
Aksi Aktor	Reaksi Sistem
1. Memasukkan username dan password.	2. Form Login akan menampilkan textbox username, password dan untuk password ditampilkan dalam bentuk kode ‘ ’ pada layar untuk jaminan keamanan.
3. Aktor melakukan konfirmasi persetujuan terhadap username, password yang telah dimasukkan dengan menekan tombol Login.	4. Aplikasi melakukan validasi terhadap username, password yang telah dimasukkan oleh pengguna dengan melakukan pengecekan pada basis data.
Kondisi Akhir	Jika pada akhir interaksi username, password yang dimasukkan pengguna valid maka pengguna akan langsung masuk ke halaman utama dan dapat menggunakan aplikasi sesuai hak aksesnya.
Alur Alternatif kondisi akhir	
a. Jika tidak valid maka Aplikasi akan kembali ke tampilan awal login.	

2. Skenario Use Case Login Chef

Skenario dalam diagram use case Login ada pada tabel seperti berikut ini:

Tabel 3.8 Skenario Use Case Login Chef

Identifikasi	
Nomor	UC2
Nama	<i>Login Chef</i>
Tujuan	Melakukan <i>Login</i> untuk masuk halaman utama.
Deskripsi	<i>Login</i> dengan memasukkan <i>Username</i> dan <i>Password</i>
Aktor	<i>Chef</i>
Skenario Utama	
Kondisi Awal	Menampilkan <i>Form Login</i> .
Aksi Aktor	<b>Reaksi Sistem</b>
1. Memasukkan <i>username</i> dan <i>password</i> .	2. <i>Form Login</i> akan menampilkan <i>textbox username, password</i> dan untuk <i>password</i> ditampilkan dalam bentuk kode ' ' pada layar untuk jaminan keamanan.
3. Aktor melakukan konfirmasi persetujuan terhadap <i>username, password</i> yang telah dimasukkan dengan menekan tombol <i>Login</i> .	4. Aplikasi melakukan validasi terhadap <i>username, password</i> yang telah dimasukkan oleh pengguna dengan melakukan pengecekan pada basis data.
Kondisi Akhir	Jika pada akhir interaksi <i>username, password</i> yang dimasukan pengguna <i>valid</i> maka pengguna akan langsung masuk ke halaman utama dan dapat menggunakan aplikasi sesuai hak aksesnya.
Alur Alternatif kondisi akhir	
a. Jika tidak <i>valid</i> maka Aplikasi akan kembali ke tampilan awal <i>login</i> .	

### 3. Skenario *Use Case Login Staf*

Skenario dalam diagram *use case Login* ada pada tabel seperti berikut ini:

*Tabel 3.9 Skenario Use Case Login Staf*

Identifikasi	
Nomor	UC3
Nama	<i>Login Staf</i>
Tujuan	Melakukan <i>Login</i> untuk masuk halaman utama.

<b>Deskripsi</b>	<i>Login dengan memasukan Username dan Password</i>
<b>Aktor</b>	Staf
<b>Skenario Utama</b>	
<b>Kondisi Awal</b>	Menampilkan <i>Form Login</i> .
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
1. Memasukkan <i>username</i> dan <i>password</i> .	2. <i>Form Login</i> akan menampilkan <i>textbox username, password</i> dan untuk <i>password</i> ditampilkan dalam bentuk kode ‘ ’ pada layar untuk jaminan keamanan.
3. Aktor melakukan konfirmasi persetujuan terhadap <i>username, password</i> yang telah dimasukkan dengan menekan tombol <i>Login</i> .	4. Aplikasi melakukan validasi terhadap <i>username, password</i> yang telah dimasukkan oleh pengguna dengan melakukan pengecekan pada basis data.
<b>Kondisi Akhir</b>	Jika pada akhir interaksi <i>username, password</i> yang dimasukan pengguna <i>valid</i> maka pengguna akan langsung masuk ke halaman utama dan dapat menggunakan aplikasi sesuai hak aksesnya.
<b>Alur Alternatif kondisi akhir</b>	
a. Jika tidak <i>valid</i> maka Aplikasi akan kembali ke tampilan awal <i>login</i> .	

#### 4. Skenario Use Case Login Pelanggan

Skenario dalam diagram *use case Login* ada pada tabel seperti berikut ini:

*Tabel 3.10 Skenario Use Case Login Pelanggan*

Identifikasi	
<b>Nomor</b>	UC4
<b>Nama</b>	<i>Login Pelanggan</i>
<b>Tujuan</b>	Melakukan <i>Login</i> untuk masuk halaman utama.
<b>Deskripsi</b>	<i>Login dengan memasukan Username dan Password</i>
<b>Aktor</b>	Pelanggan
<b>Skenario Utama</b>	
<b>Kondisi Awal</b>	Menampilkan <i>Form Login</i> .
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
1. Memasukkan <i>username</i> dan <i>password</i> .	2. <i>Form Login</i> akan menampilkan <i>textbox username, password</i> dan untuk <i>password</i> ditampilkan dalam bentuk kode ‘ ’ pada layar untuk jaminan keamanan.

3. Aktor melakukan konfirmasi persetujuan terhadap <i>username, password</i> yang telah dimasukkan dengan menekan tombol <i>Login</i> .	4. Aplikasi melakukan validasi terhadap <i>username, password</i> yang telah dimasukkan oleh pengguna dengan melakukan pengecekan pada basis data.
<b>Kondisi Akhir</b>	Jika pada akhir interaksi <i>username, password</i> yang dimasukan pengguna <i>valid</i> maka pengguna akan langsung masuk ke halaman utama dan dapat menggunakan aplikasi sesuai hak aksesnya.
<b>Alur Alternatif kondisi akhir</b> a. Jika tidak <i>valid</i> maka Aplikasi akan kembali ke tampilan awal <i>login</i> .	

## 5. Skenario *Use Case Login Kelola Data Makanan dan Minuman*

Skenario dalam diagram *use case* Kelola Data Makanan dan Minuman ada pada tabel seperti berikut ini:

*Tabel 3.11 Skenario Use Case Kelola Data Makanan dan Minuman*

Identifikasi	
Nomor	UC5
Nama	Kelola Data Makanan dan Minuman
Tujuan	Melakukan manipulasi isi <i>record</i> data
Deskripsi	Melakukan CRUD terhadap data menu makanan. Dan minuman
Aktor	Admin
Skenario Utama	
Kondisi Awal	<i>Form login</i> terlebih dahulu. Jika <i>valid</i> maka masuk ke <i>form</i> halaman utama admin jika tidak <i>valid</i> akan kembali ke tampilan awal <i>login</i> .
Aksi Aktor	<b>Reaksi Sistem</b>
1. Admin memilih <i>link</i> ke halaman yang akan dimanipulasi.	2. Aplikasi menampilkan <i>form</i> sesuai pilihan pengguna.
3. Admin melakukan manipulasi isi <i>record</i> dengan memilih tombol add, edit atau delete.	4. Aplikasi menampilkan pesan konfirmasi perintah untuk menyimpan data hasil penambahan, perubahan atau penghapusan data makanan dan minuman.
5. Admin melakukan konfirmasi persetujuan manipulasi isi <i>record</i> .	6. Aplikasi menyimpan hasil manipulasi pada basis data tabel <i>tbl_produk</i> atau yang terkait.
Kondisi Akhir	Isi informasi <i>record</i> pada basis data tabel

	tbl_produk yang dilakukan manipulasi mengalami perubahan.
<b>Alur alternatif no.4</b> a. Jika data yang dimasukkan tidak sesuai dengan tipe data yang bisa diterima basis data maka akan dimunculkan pesan <i>error</i> . b. Jika ada <i>textbox</i> yang wajib diisi masih kosong maka akan muncul pesan peringatan untuk melakukan pengecekan ulang.	

## 6. Skenario Use Case Kelola Data User

Skenario dalam diagram *use case* Kelola Data User ada pada tabel seperti berikut ini:

*Tabel 3.12 Skenario Use Case Kelola Data User*

Identifikasi	
Nomor	UC6
Nama	Kelola Data User
Tujuan	Melakukan manipulasi isi <i>record</i> data
Deskripsi	Melakukan CRUD terhadap data <i>user</i>
Aktor	Admin
Skenario Utama	
Kondisi Awal	<i>Form login</i> terlebih dahulu. Jika <i>valid</i> maka masuk ke <i>form</i> halaman utama admin jika tidak <i>valid</i> akan kembali ke tampilan awal <i>login</i> .
Aksi Aktor	Reaksi Sistem
1. Admin memilih <i>link</i> ke halaman yang akan dimanipulasi.	2. Aplikasi menampilkan <i>form</i> sesuai pilihan pengguna.
3. Admin melakukan manipulasi isi <i>record</i> dengan memilih tombol tambah, edit atau hapus.	4. Aplikasi menampilkan pesan konfirmasi perintah untuk menyimpan data hasil penambahan, perubahan atau penghapusan data <i>user</i> .
5. Admin melakukan konfirmasi persetujuan manipulasi isi <i>record</i> .	6. Aplikasi menyimpan hasil manipulasi pada basis data tabel <i>tbl_user</i> atau yang terkait.
Kondisi Akhir	Isi informasi <i>record</i> pada basis data tabel <i>tbl_user</i> yang dilakukan manipulasi mengalami perubahan.
<b>Alur alternatif no.4</b> a. Jika data yang dimasukkan tidak sesuai dengan tipe data yang bisa diterima basis data maka akan dimunculkan pesan <i>error</i> . b. Jika ada <i>textbox</i> yang wajib diisi masih kosong maka akan muncul pesan peringatan untuk melakukan pengecekan ulang	



## 7. Skenario *Use Case* Kelola Status Masakan

Skenario dalam diagram *use case* Kelola Status Masakan ada pada tabel seperti berikut ini:

*Tabel 3.13 Skenario Use Case Kelola Status Masakan*

Identifikasi	
Nomor	UC7
Nama	Kelola Status Masakan
Tujuan	Melakukan manipulasi isi <i>record</i> data
Deskripsi	Melakukan <i>update</i> terhadap status menjadi “selesai”
Aktor	<i>Chef</i>
Skenario Utama	
Kondisi Awal	<i>Form login</i> terlebih dahulu. Jika <i>valid</i> maka masuk ke <i>form</i> halaman utama admin jika tidak <i>valid</i> akan kembali ke tampilan awal <i>login</i> .
Aksi Aktor	Reaksi Sistem
1. <i>Chef</i> memilih <i>link</i> ke halaman yang akan dimanipulasi.	2. Aplikasi menampilkan <i>form</i> sesuai pilihan pengguna.
3. <i>Chef</i> melakukan manipulasi isi <i>record</i> dengan memilih tombol <i>update</i> status.	4. Aplikasi menampilkan pesan konfirmasi perintah untuk menyimpan data hasil perubahan status pada <i>field</i> status.
5. <i>Chef</i> melakukan konfirmasi persetujuan manipulasi isi <i>record</i> .	6. Aplikasi menyimpan hasil manipulasi pada basis data tabel <i>tbl_pemesanan</i> atau yang terkait.
Kondisi Akhir	Isi informasi <i>record</i> pada basis data tabel <i>tbl_pemesanan</i> yang dilakukan manipulasi mengalami perubahan.

## 8. Skenario *Use Case* Kelola Status Pengantaran

Skenario dalam diagram *use case* Kelola Status Pengantaran ada pada tabel seperti berikut ini:

*Tabel 3.14 Skenario Use Case Kelola Status Pengantaran*

Identifikasi	
Nomor	UC8
Nama	Kelola Status Pengantaran
Tujuan	Melakukan manipulasi isi <i>record</i> data
Deskripsi	Melakukan <i>update</i> terhadap status menjadi “diantar”
Aktor	Staf
Skenario Utama	

<b>Kondisi Awal</b>	<i>Form login</i> terlebih dahulu. Jika <i>valid</i> maka masuk ke <i>form</i> halaman utama untuk kasir jika tidak <i>valid</i> akan kembali ke tampilan awal <i>login</i> .
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
1. Staf memilih <i>link</i> ke halaman yang akan dimanipulasi.	2. Aplikasi menampilkan <i>form</i> sesuai pilihan pengguna.
3. Staf melakukan manipulasi isi <i>record</i> dengan memilih tombol tambah, edit atau hapus.	4. Aplikasi menampilkan pesan konfirmasi perintah untuk menyimpan data hasil perubahan status pada <i>field</i> status.
5. Staf melakukan konfirmasi persetujuan manipulasi isi <i>record</i> .	6. Aplikasi menyimpan hasil manipulasi pada basis data tabel <i>tbl_pemesanan</i> atau yang terkait.
<b>Kondisi Akhir</b>	Isi informasi <i>record</i> pada basis data tabel <i>tbl_pemesanan</i> yang dilakukan manipulasi mengalami perubahan.

## 9. Skenario Use Case View Data Makanan dan Minuman

Skenario dalam diagram *use case View* Data Makanan dan Minuman ada pada tabel seperti berikut ini:

*Tabel 3.15 Skenario Use Case View Data Makanan dan Minuman*

Identifikasi	
<b>Nomor</b>	UC9
<b>Nama</b>	Data Makanan dan Minuman
<b>Tujuan</b>	Melihat isi <i>record</i> data
<b>Deskripsi</b>	<i>View record</i> data yang telah diinputkan
<b>Aktor</b>	Admin, <i>Chef</i> , Pelanggan
<b>Skenario Utama</b>	
<b>Kondisi Awal</b>	<i>Form login</i> terlebih dahulu. Jika <i>valid</i> maka masuk ke <i>form</i> halaman utama untuk kasir jika tidak <i>valid</i> akan kembali ke tampilan awal <i>login</i> .
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
1. Staf memilih <i>link</i> ke halaman yang akan dilihat.	2. Aplikasi menampilkan data sesuai pilihan pengguna.
<b>Kondisi Akhir</b>	Isi informasi <i>record</i> pada basis data tabel <i>tbl_produk</i> ditampilkan.

## 10. Skenario Use Case Input Data Pesanan

Skenario dalam diagram *use case* Input Data Pesanan

ada pada tabel seperti berikut ini:

*Tabel 3.16 Skenario Use Case Input Data Pesanan*

Identifikasi	
Nomor	UC10
Nama	Input Data Pesanan
Tujuan	Menambahkan isi <i>record</i> data
Deskripsi	Menambahkan <i>record</i> ke tabel <i>tbl_pemesanan</i>
Aktor	Pelanggan
Skenario Utama	
Kondisi Awal	<i>Form login</i> terlebih dahulu. Jika <i>valid</i> maka masuk ke <i>form</i> halaman utama untuk kasir jika tidak <i>valid</i> akan kembali ke tampilan awal <i>login</i> .
Aksi Aktor	Reaksi Sistem
1. Pelanggan memilih <i>link</i> ke halaman yang akan dimanipulasi.	2. Aplikasi menampilkan tampilan sesuai pilihan pengguna.
3. Pelanggan melakukan pemilihan dan menginputkan jumlah kursi, estimasi kedatangan, jumlah produk yang akan dibeli.	4. Aplikasi menampilkan pesan konfirmasi perintah untuk menyimpan data hasil penambahan pada tabel <i>tbl_pemesanan</i> .
5. Pelanggan melakukan konfirmasi persetujuan menambah isi <i>record</i> .	6. Aplikasi menyimpan hasil penambahan pada basis data tabel <i>tbl_pemesanan</i> atau yang terkait.
Kondisi Akhir	Isi informasi <i>record</i> pada basis data tabel <i>tbl_pemesanan</i> yang dilakukan penambahan mengalami perubahan.

## 11. Skenario Use Case View Data Pesanan

Skenario dalam diagram *use case* Kelola Status

Pengantaran ada pada tabel seperti berikut ini:

*Tabel 3.17 Skenario Use Case View Data Pesanan*

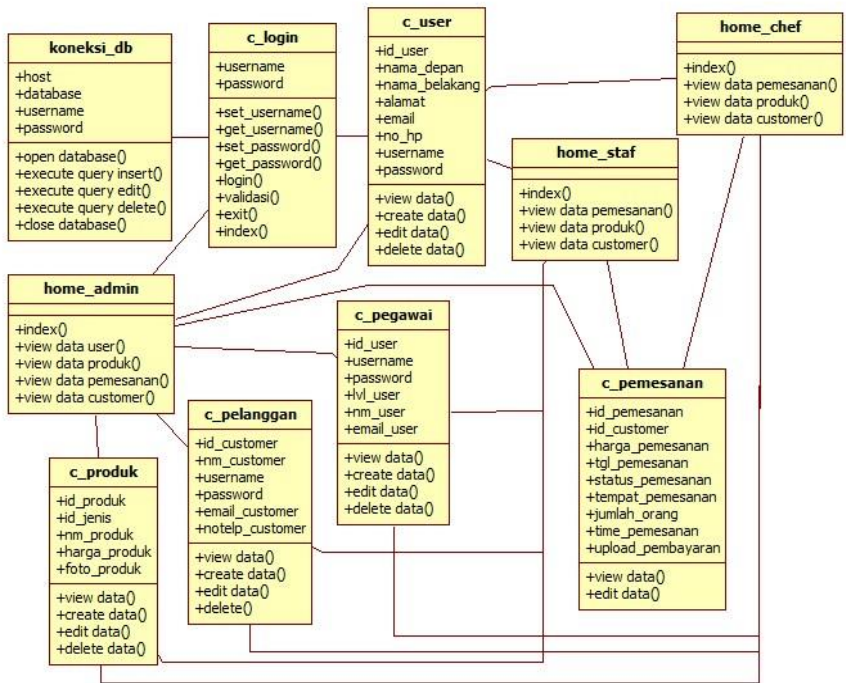
Identifikasi	
Nomor	UC11
Nama	View Data Pesanan
Tujuan	Melihat isi <i>record</i> data
Deskripsi	<i>View record</i> data pesanan yang telah diinputkan
Aktor	Admin, <i>Chef</i> , Staf, Pelanggan
Skenario Utama	
Kondisi Awal	<i>Form login</i> terlebih dahulu. Jika <i>valid</i> maka masuk ke <i>form</i> halaman utama untuk kasir

	jika tidak <i>valid</i> akan kembali ke tampilan awal <i>login</i> .
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
1. Admin, <i>Chef</i> , Staf, Pelanggan memilih <i>link</i> ke halaman yang akan dilihat.	2. Aplikasi menampilkan tampilan sesuai pilihan pengguna.
<b>Kondisi Akhir</b>	Isi informasi <i>record</i> pada basis data tabel <i>tbl_pemesanan</i> ditampilkan.

### 3.2.2 Class Diagram

*Class diagram* adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek [6].

Dalam proses pembuatan aplikasi ini *class diagram* yang akan digunakan adalah sebagai berikut:

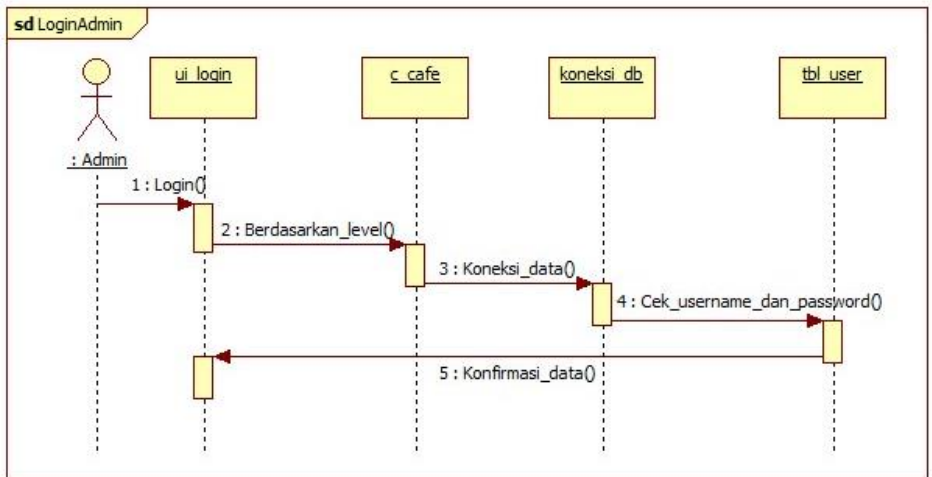


Gambar 3. 5 Class Diagram

### 3.2.3 Sequence Diagram

Sequence diagram adalah suatu diagram yang menggambarkan interaksi antar objek dan mengindikasikan komunikasi diantara objek-objek di dalam dan disekitar sistem [7].

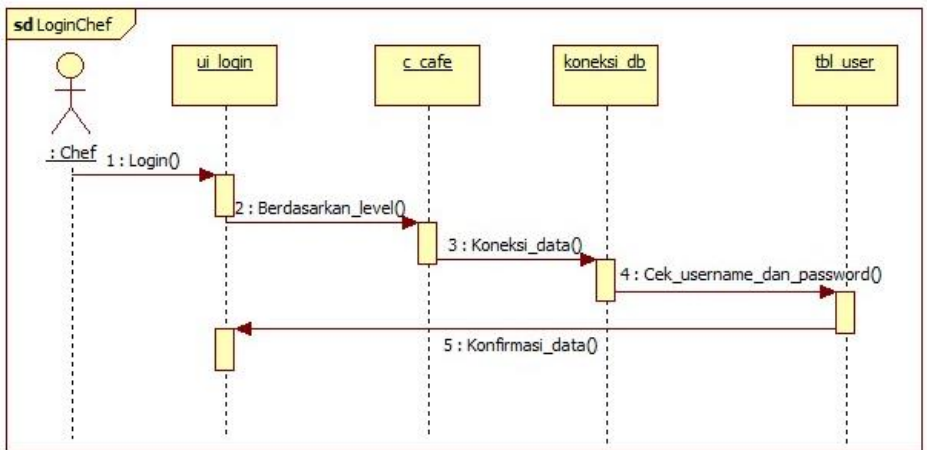
### 3.2.3.1 Sequence Diagram Login Admin (UC1)



Gambar 3. 6 Sequence Diagram Login Admin (UC1)

Pada *sequence diagram* ini menjelaskan proses *login*. Aktor mulai menjalankan aplikasi dan akan tampil *form login*. Kemudian aktor memasukkan *username* dan *password*, pada *interface login* data yang di *input* akan di cek di tabel `tbl_user` di *database*. Jika sesuai maka akan mendapatkan tampilan form utama ke pengguna.

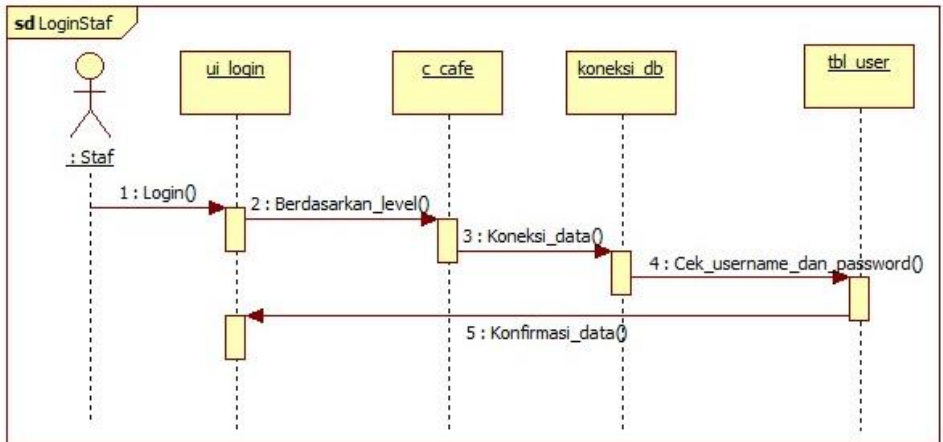
### 3.2.3.2 Sequence Diagram Login Chef (UC2)



Gambar 3. 7 Sequence Diagram Login Chef (UC2)

Pada *sequence diagram* ini menjelaskan proses *login*. Aktor mulai menjalankan aplikasi dan akan tampil *form login*. Kemudian aktor memasukkan *username dan password*, pada *interface login* data yang di *input* akan di cek di tabel *tbl\_user* di *database*. Jika sesuai maka akan mendapatkan tampilan form utama ke pengguna.

### 3.2.3.3 Sequence Diagram Login Staf (UC3)



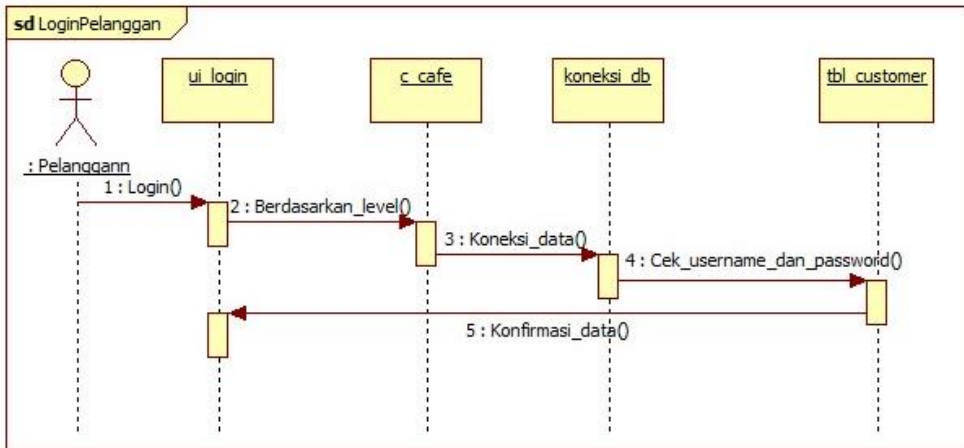
*Gambar 3. 8 Sequence Diagram Login Staf (UC3)*

Pada *sequence diagram* ini menjelaskan proses *login*.

Aktor mulai menjalankan aplikasi dan akan tampil *form login*. Kemudian aktor memasukkan *username dan password*, pada *interface login* data yang di input akan di cek di tabel *tbl\_user* di *database*. Jika sesuai maka akan mendapatkan tampilan form utama ke pengguna.



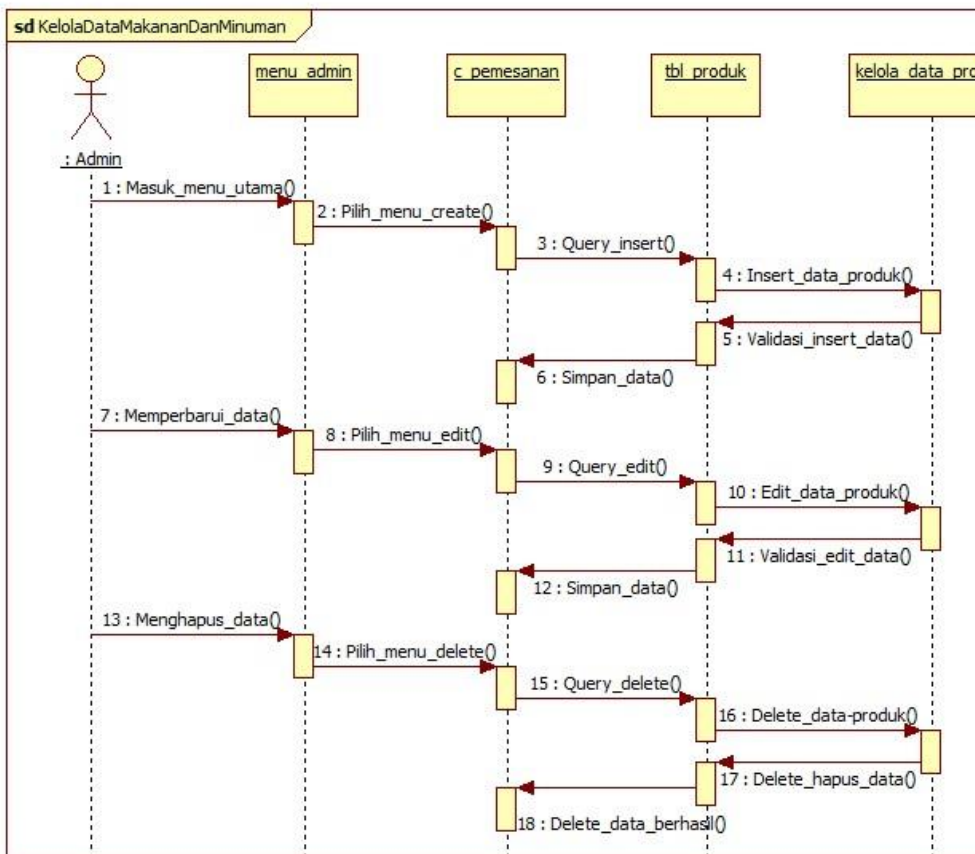
### 3.2.3.4 Sequence Diagram Login Pelanggan (UC4)



Gambar 3. 9 Sequence Diagram Login Pelanggan (UC4)

Pada *sequence diagram* ini menjelaskan proses *login*. Aktor mulai menjalankan aplikasi dan akan tampil *form login*. Kemudian aktor memasukkan *username* dan *password*, pada *interface login* data yang di input akan di cek di tabel *tbl\_customer* di *database*. Jika sesuai maka akan mendapatkan tampilan form utama ke pengguna.

### 3.2.3.5 Sequence Diagram Kelola Data Makanan dan Minuman (UC5)

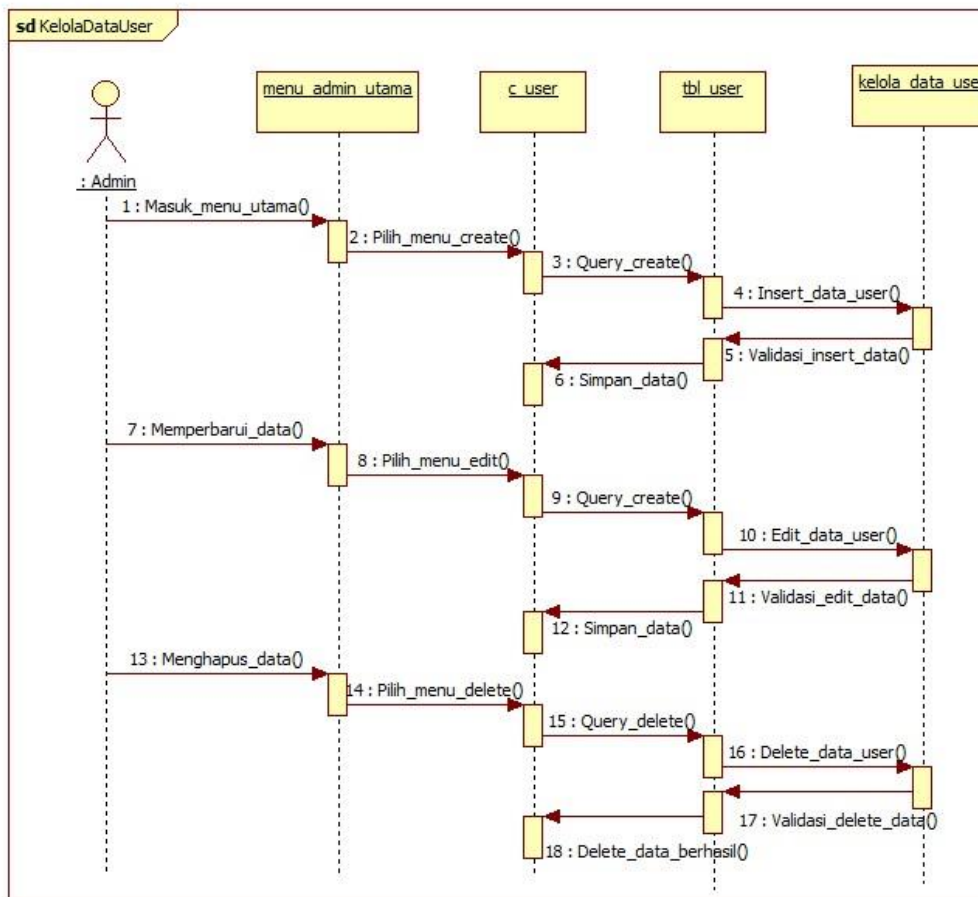


Gambar 3. 10 Sequence Diagram Kelola Data Makanan dan Minuman (UC5)

Pada *sequence diagram* ini menjelaskan proses kelola data makanan dan minuman. Aktor melakukan *login* terlebih dahulu. Selanjutnya sistem akan memuat *form* menu utama. Untuk memulai kelola ini, masuk ke menu kelola data makanan dan minuman, kemudian sistem akan melakukan

proses koneksi *database* dan dilanjutkan dengan proses *OpenTable* ke tabel yang dituju yaitu *tbl\_produk* dengan melakukan operasi *insert()*, *update()*, *delete()*. Selanjutnya sistem akan memvalidasi hasil dari operasi, apakah berhasil atau gagal. Dan yang terakhir sistem akan melakukan proses *CloseDB* untuk menutup koneksi.

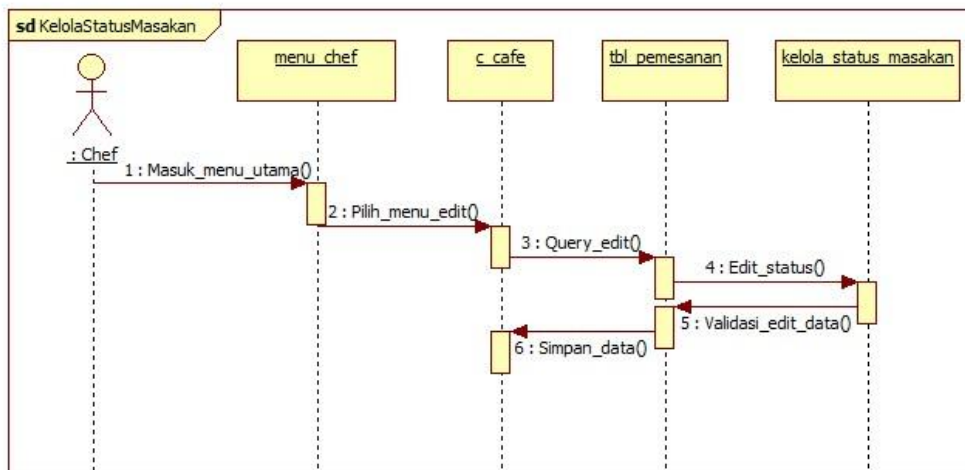
### 3.2.3.6 Sequence Diagram Kelola Data User (UC6)



Gambar 3. 11 Sequence Diagram Kelola Data User (UC6)

Pada *sequence diagram* ini menjelaskan proses kelola data *user*. Aktor melakukan *login* terlebih dahulu. Selanjutnya sistem akan memuat *form* menu utama. Untuk memulai kelola ini, masuk ke menu kelola data *user*, kemudian sistem akan melakukan proses koneksi database dan dilanjutkan dengan proses *OpenTable* ke tabel yang dituju yaitu *tbl\_user* dengan melakukan operasi *insert()*, *update()*, *delete()*. Selanjutnya sistem akan memvalidasi hasil dari operasi, apakah berhasil atau gagal. Dan yang terakhir sistem akan melakukan proses *CloseDB* untuk menutup koneksi.

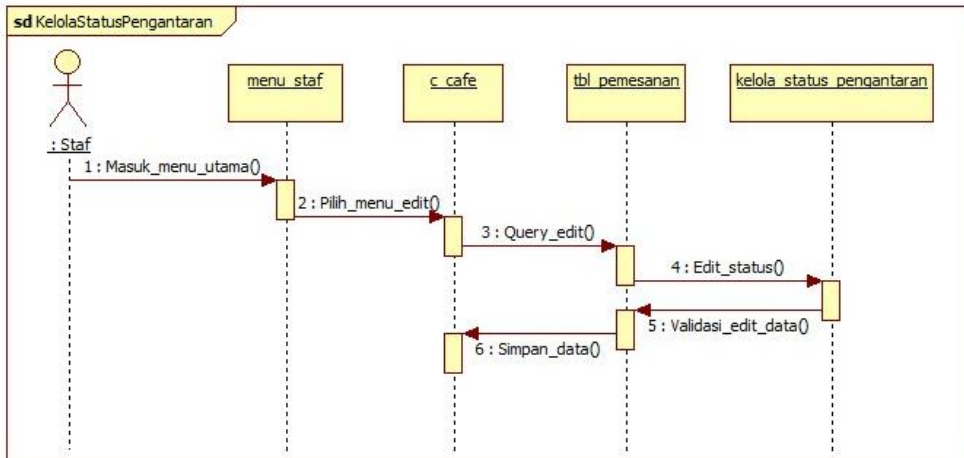
### 3.2.3.7 Sequence Diagram Kelola Status Masakan (UC7)



Gambar 3. 12 Sequence Diagram Kelola Status Masakan (UC7)

Pada *sequence diagram* ini menjelaskan proses kelola status masakan. Aktor melakukan *login* terlebih dahulu. Selanjutnya sistem akan memuat *form* menu utama. Untuk memulai kelola ini, masuk ke menu kelola status pesanan, kemudian sistem akan melakukan proses koneksi *database* dan dilanjutkan dengan proses *OpenTable* ke tabel yang dituju yaitu *tbl\_pemesanan* dengan melakukan operasi *update()*. Selanjutnya sistem akan memvalidasi hasil dari operasi, apakah berhasil atau gagal. Dan yang terakhir sistem akan melakukan proses *CloseDB* untuk menutup koneksi.

### 3.2.3.8 Sequence Diagram Kelola Status Pengantaran (UC8)

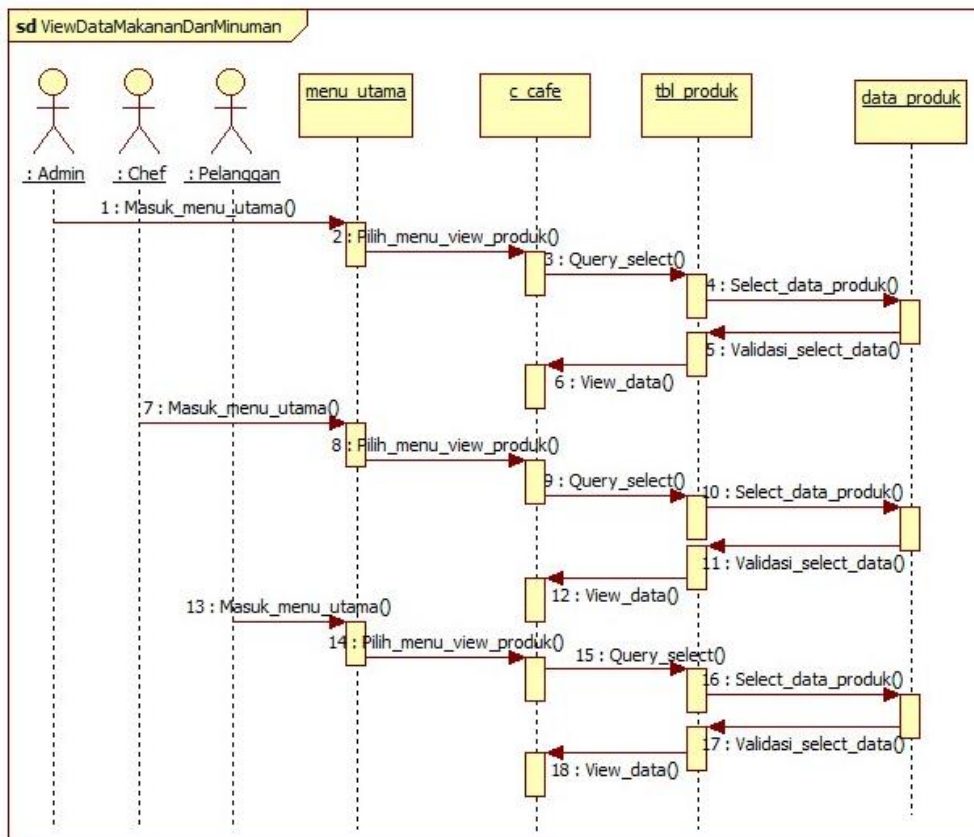


Gambar 3. 13 Sequence Diagram Kelola Status Pengantaran (UC8)

Pada *sequence diagram* ini menjelaskan proses kelola status pengantaran pesanan. Aktor melakukan *login* terlebih dahulu. Selanjutnya sistem akan memuat *form* menu utama. Untuk memulai kelola ini, masuk ke menu kelola status

pengantaran pesanan, kemudian sistem akan melakukan proses koneksi *database* dan dilanjutkan dengan proses *OpenTable* ke tabel yang dituju yaitu *tbl\_pemesanan* dengan melakukan operasi *update()*. Selanjutnya sistem akan memvalidasi hasil dari operasi, apakah berhasil atau gagal. Dan yang terakhir sistem akan melakukan proses *CloseDB* untuk menutup koneksi.

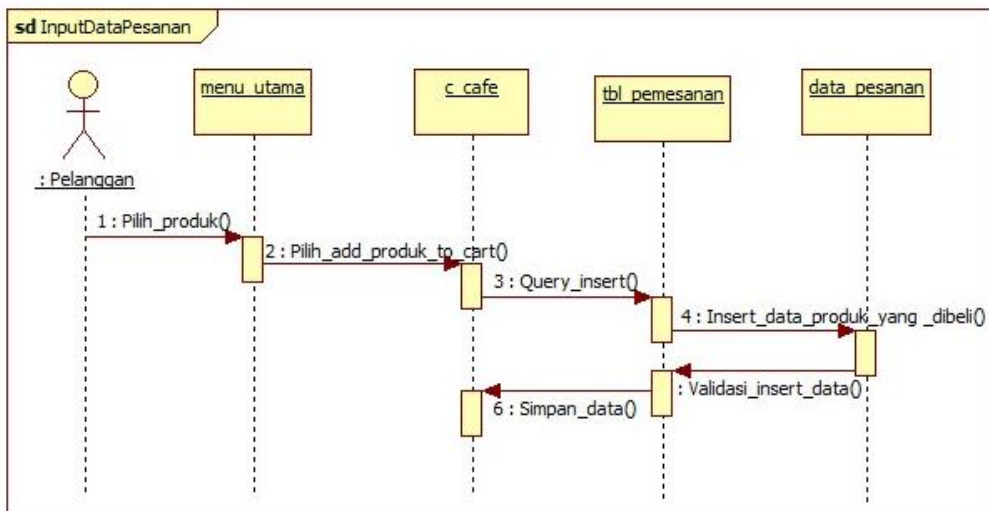
### 3.2.3.9 Sequence Diagram View Data Makanan dan Minuman (UC9)



Gambar 3. 14 Sequence Diagram View Data Makanan dan Minuman (UC9)

Pada *sequence diagram* ini menjelaskan proses *view* data makanan dan minuman. Aktor melakukan *login* terlebih dahulu. Selanjutnya sistem akan memuat *form* menu utama. Untuk memulai proses ini, masuk ke menu data produk, kemudian sistem akan melakukan proses koneksi *database* dan dilanjutkan dengan proses *OpenTable* ke tabel yang dituju yaitu *tbl\_produk* dengan melakukan operasi *select()*. Selanjutnya sistem akan memvalidasi hasil dari operasi, apakah berhasil atau gagal. Dan yang terakhir sistem akan melakukan proses *CloseDB* untuk menutup koneksi.

### 3.2.3.10 Sequence Diagram Input Data Pesanan (UC10)

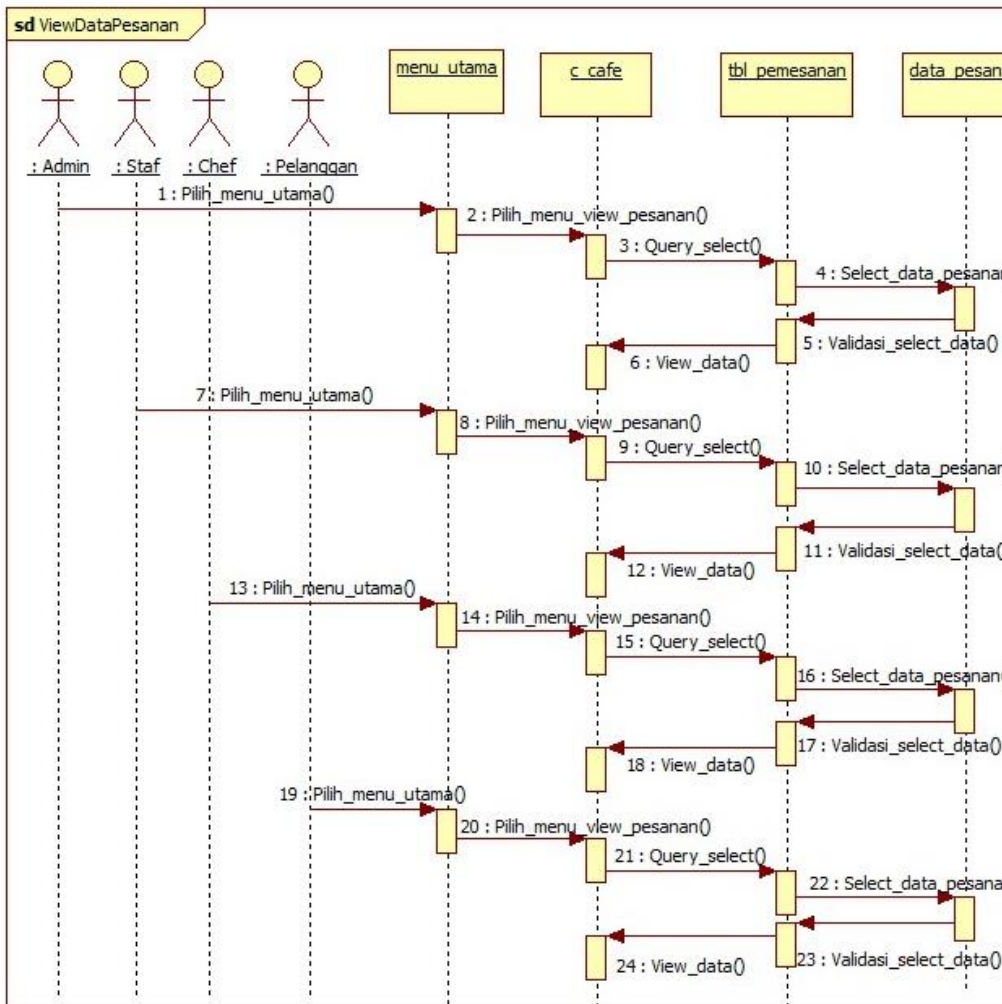


Gambar 3. 15 Sequence Diagram Input Data Pesanan (UC10)

Pada *sequence diagram* ini menjelaskan proses input data pesanan. Aktor melakukan *login* terlebih dahulu. Selanjutnya sistem akan memuat *form* menu utama. Untuk memulai proses ini, masuk ke menu pemesanan, kemudian sistem akan melakukan proses koneksi *database* dan dilanjutkan dengan proses *OpenTable* ke tabel yang dituju yaitu `tbl_pemesanan` dengan melakukan operasi *insert()*. Selanjutnya sistem akan memvalidasi hasil dari operasi, apakah berhasil atau gagal. Dan yang terakhir sistem akan melakukan proses *CloseDB* untuk menutup koneksi.



### 3.2.3.11 Sequence Diagram View Data Pesanan (UC11)



Gambar 3. 16 Sequence Diagram View Data Pesanan  
(UC11)

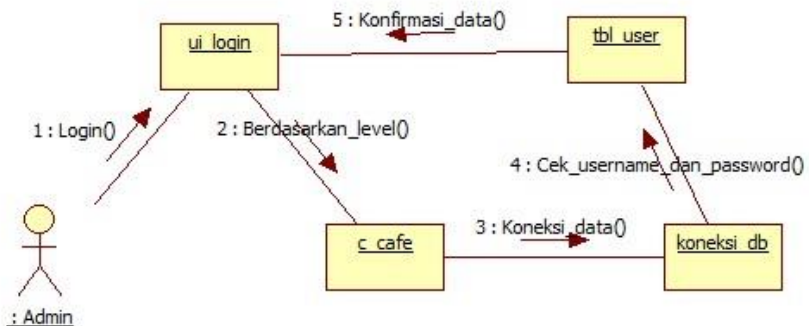
Pada *sequence diagram* ini menjelaskan proses melihat data pesanan. Aktor melakukan *login* terlebih dahulu. Selanjutnya sistem akan memuat *form* menu utama. Untuk

memulai proses ini, masuk ke menu pesanan, kemudian sistem akan melakukan proses koneksi *database* dan dilanjutkan dengan proses *OpenTable* ke tabel yang dituju yaitu *tbl\_pemesanan* dengan melakukan operasi *select()*. Selanjutnya sistem akan memvalidasi hasil dari operasi, apakah berhasil atau gagal. Dan yang terakhir sistem akan melakukan proses *CloseDB* untuk menutup koneksi.

### 3.2.4 Communication Diagram

*Communication diagram* adalah sebuah interaksi yang memberi tekanan pada hubungan-hubungan data antar partisipan yang berbeda dalam sebuah interaksi.

#### 3.2.4.1 Communication Diagram Login Admin (UC1)

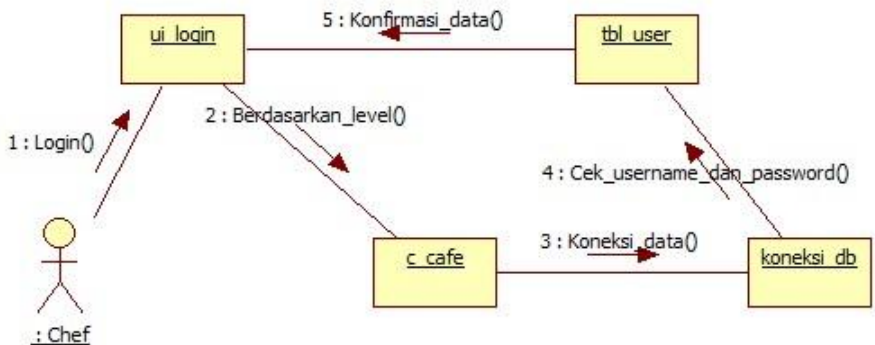


Gambar 3. 17 Communication Diagram Login Admin (UC1)

Pada *communication diagram* ini menjelaskan proses *login* admin. Aktor mulai menjalankan aplikasi dan akan tampil *form login*. Kemudian aktor memasukkan *username* dan *password*, pada *interface login* data yang di *input* akan di

cek di tabel `tbl_user` di *database*. Jika sesuai maka akan mendapatkan tampilan *form* utama ke pengguna.

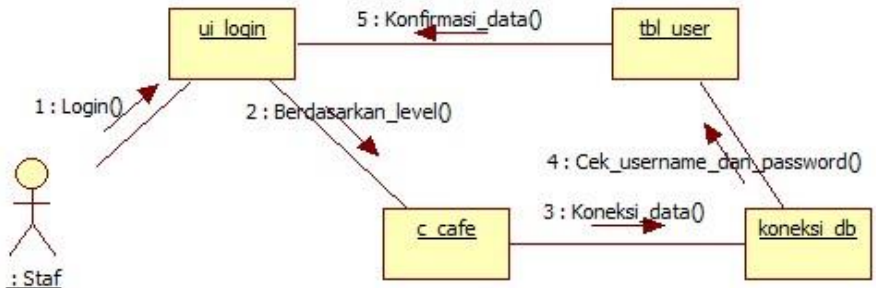
#### 3.2.4.2 Communication Diagram Login Chef (UC2)



Gambar 3. 18 Communication Diagram Login Chef (UC2)

Pada *communication* diagram ini menjelaskan proses *login chef*. Aktor mulai menjalankan aplikasi dan akan tampil *form login*. Kemudian aktor memasukkan *username* dan *password*, pada *interface login* data yang di *input* akan di cek di tabel `tbl_user` di *database*. Jika sesuai maka akan mendapatkan tampilan *form* utama ke pengguna.

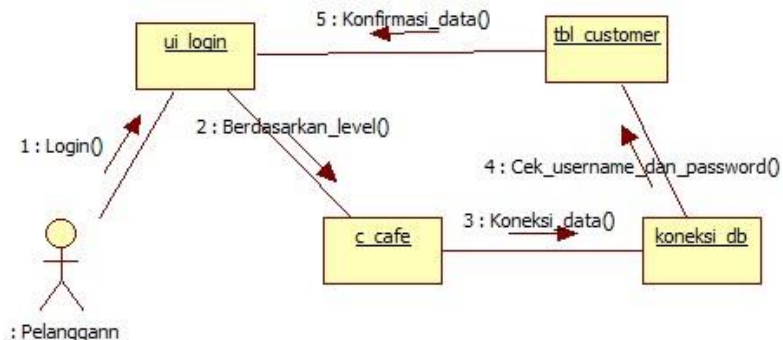
### 3.2.4.3 Communication Diagram Login Staf (UC3)



Gambar 3. 19 Communication Diagram Login Staf (UC3)

Pada *communication* diagram ini menjelaskan proses *login chef*. Aktor mulai menjalankan aplikasi dan akan tampil *form login*. Kemudian aktor memasukkan *username* dan *password*, pada *interface login* data yang di *input* akan di cek di tabel `tbl_user` di *database*. Jika sesuai maka akan mendapatkan tampilan *form* utama ke pengguna.

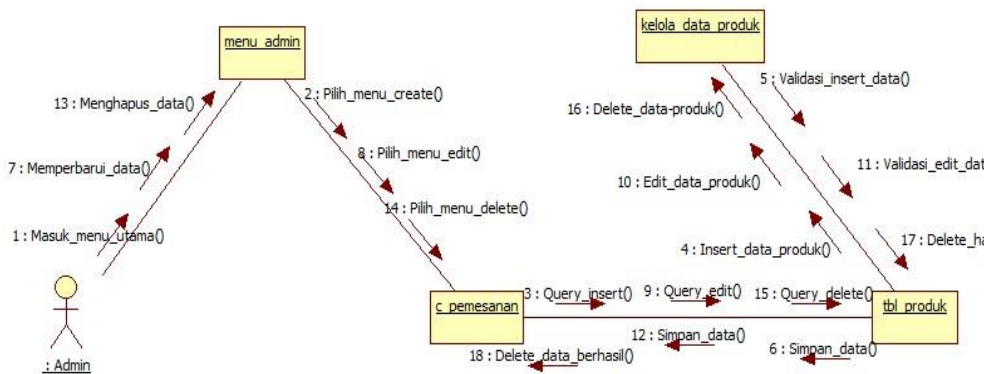
### 3.2.4.4 Communication Diagram Login Pelanggan (UC4)



Gambar 3. 20 Communication Diagram Login Pelanggan (UC4)

Pada *communication* diagram ini menjelaskan proses *login* pelayan. Aktor mulai menjalankan aplikasi dan akan tampil *form login*. Kemudian aktor memasukkan *username* dan *password*, pada *interface login* data yang di *input* akan di cek di tabel *tbl\_user* di *database*. Jika sesuai maka akan mendapatkan tampilan *form* utama ke pengguna.

### 3.2.4.5 Communication Diagram Kelola Data Makanan dan Minuman (UC5)

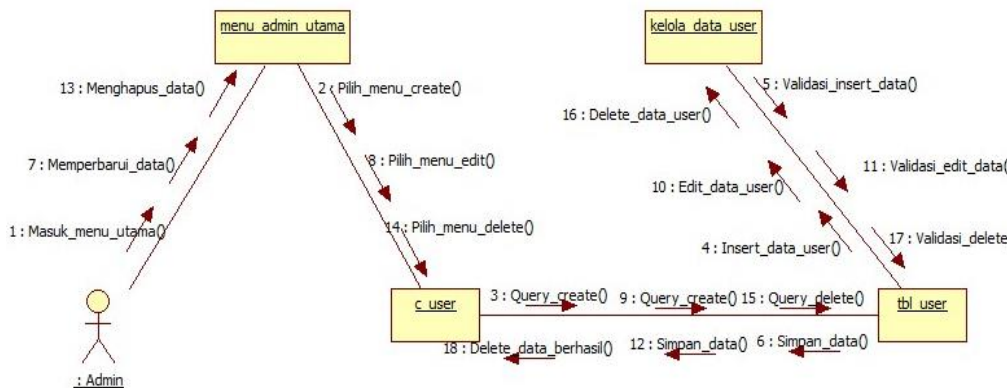


Gambar 3. 21 Communication Diagram Kelola Data Makanan dan Minuman (UC5)

Pada *communication* diagram ini menjelaskan proses kelola data makanan dan minuman. Aktor melakukan *login* terlebih dahulu. Selanjutnya sistem akan memuat *form* menu utama. Untuk memulai kelola ini, masuk ke menu kelola data makanan dan minuman, kemudian sistem akan melakukan proses koneksi *database* dan dilanjutkan dengan proses

*OpenTable* ke tabel yang dituju yaitu *tbl\_produk* dengan melakukan operasi *insert()*, *update()*, *delete()*. Selanjutnya sistem akan memvalidasi hasil dari operasi, apakah berhasil atau gagal. Dan yang terakhir sistem akan melakukan proses *CloseDB* untuk menutup koneksi.

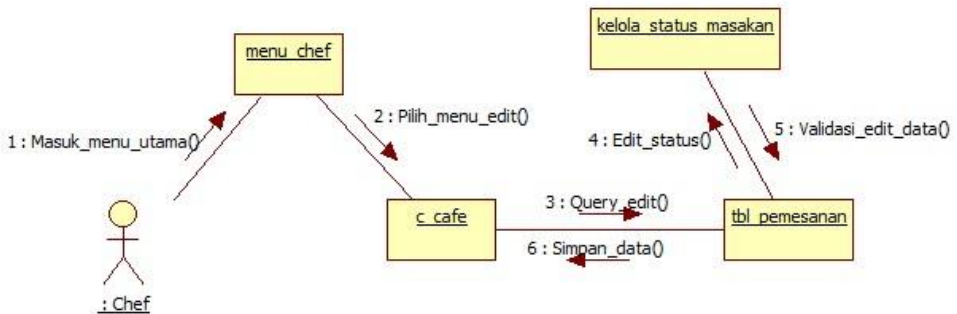
### 3.2.4.6 Communication Diagram Kelola Data User (UC6)



Gambar 3. 22 Communication Diagram Kelola Data User (UC6)

Pada *communication diagram* ini menjelaskan proses kelola data *user*. Aktor melakukan *login* terlebih dahulu. Selanjutnya sistem akan memuat *form* menu utama. Untuk memulai kelola ini, masuk ke menu kelola data *user*, kemudian sistem akan melakukan proses koneksi *database* dan dilanjutkan dengan proses *OpenTable* ke tabel yang dituju yaitu *tbl\_user* dengan melakukan operasi *insert()*, *update()*, *delete()*. Selanjutnya sistem akan memvalidasi hasil dari operasi, apakah berhasil atau gagal. Dan yang terakhir sistem akan melakukan proses *CloseDB* untuk menutup koneksi.

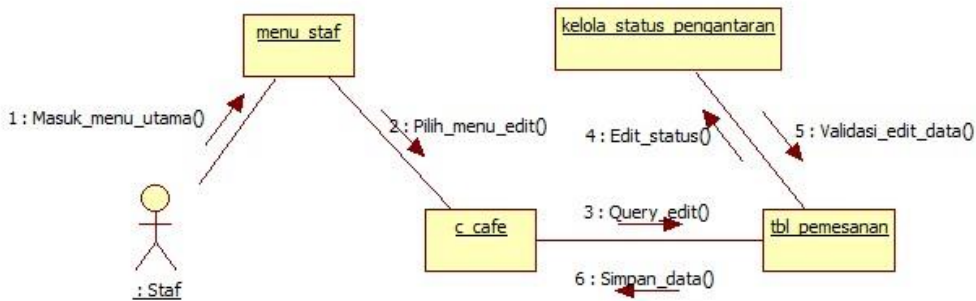
### 3.2.4.7 Communication Diagram Kelola Status Masakan (UC7)



Gambar 3. 23 Communication Diagram Kelola Status Masakan (UC7)

Pada *communication* diagram ini menjelaskan proses kelola status masakan. Aktor melakukan *login* terlebih dahulu. Selanjutnya sistem akan memuat form menu utama. Untuk memulai kelola ini, masuk ke menu kelola status pesanan, kemudian sistem akan melakukan proses koneksi *database* dan dilanjutkan dengan proses *OpenTable* ke tabel yang dituju yaitu *tbl\_pemesanan* dengan melakukan operasi *update()*. Selanjutnya sistem akan memvalidasi hasil dari operasi, apakah berhasil atau gagal. Dan yang terakhir sistem akan melakukan proses *CloseDB* untuk menutup koneksi.

### 3.2.4.8 Communication Diagram Kelola Status Pengantaran (UC8)

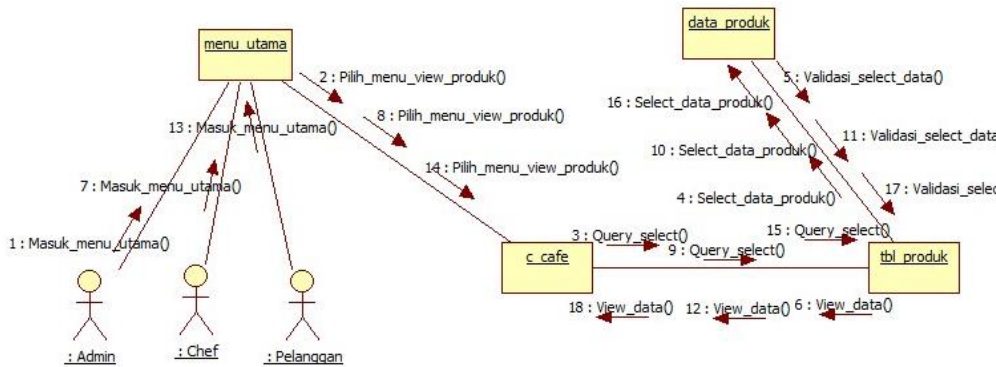


Gambar 3. 24 Communication Diagram Kelola Status Pengantaran (UC8)

Pada *communication diagram* ini menjelaskan proses kelola status pengantaran. Aktor melakukan *login* terlebih dahulu. Selanjutnya sistem akan memuat form menu utama. Untuk memulai kelola ini, masuk ke menu kelola status pesanan, kemudian sistem akan melakukan proses koneksi *database* dan dilanjutkan dengan proses *OpenTable* ke tabel yang dituju yaitu *tbl\_pemesanan* dengan melakukan operasi *update()*. Selanjutnya sistem akan memvalidasi hasil dari operasi, apakah berhasil atau gagal. Dan yang terakhir sistem akan melakukan proses *CloseDB* untuk menutup koneksi.



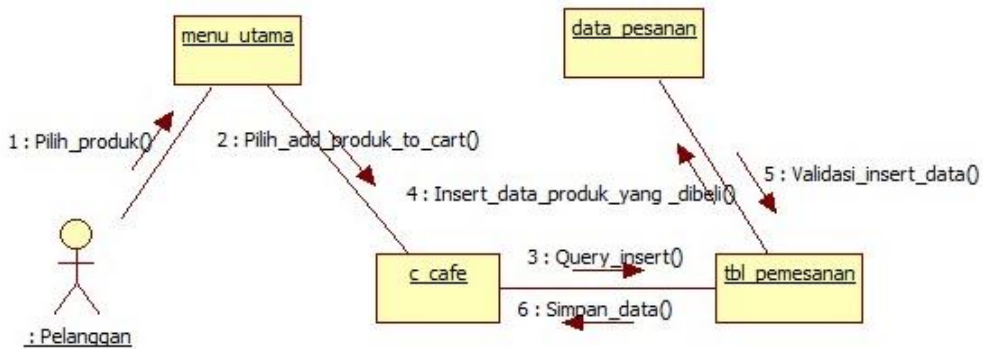
### 3.2.4.9 Communication Diagram View Data Makanan dan Minuman (UC9)



Gambar 3. 25 Communication Diagram View Data Makanan dan Minuman (UC9)

Pada *communication diagram* ini menjelaskan proses *view* data makanan dan minuman. Aktor melakukan *login* terlebih dahulu. Selanjutnya sistem akan memuat form menu utama. Untuk memulai proses ini, masuk ke menu data produk, kemudian sistem akan melakukan proses koneksi *database* dan dilanjutkan dengan proses *OpenTable* ke tabel yang dituju yaitu *tbl\_produk* dengan melakukan operasi *select()*. Selanjutnya sistem akan memvalidasi hasil dari operasi, apakah berhasil atau gagal. Dan yang terakhir sistem akan melakukan proses *CloseDB* untuk menutup koneksi.

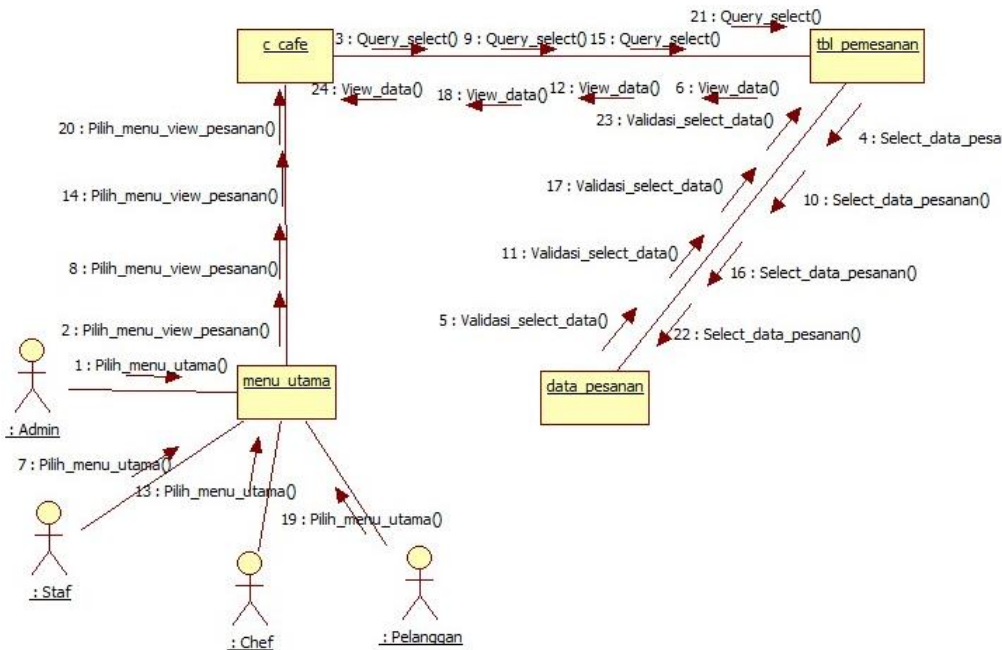
### 3.2.4.10 Communication Diagram Input Data Pesanan (UC10)



Gambar 3. 26 Communication Diagram Input Data Pesanan (UC10)

Pada *communication diagram* ini menjelaskan proses input data pesanan. Aktor melakukan *login* terlebih dahulu. Selanjutnya sistem akan memuat form menu utama. Untuk memulai proses ini, masuk ke menu pemesanan, kemudian sistem akan melakukan proses koneksi *database* dan dilanjutkan dengan proses *OpenTable* ke tabel yang dituju yaitu *tbl\_pemesanan* dengan melakukan operasi *insert()*. Selanjutnya sistem akan memvalidasi hasil dari operasi, apakah berhasil atau gagal. Dan yang terakhir sistem akan melakukan proses *CloseDB* untuk menutup koneksi.

### 3.2.4.11 Communication Diagram View Data Pesanan (UC11)



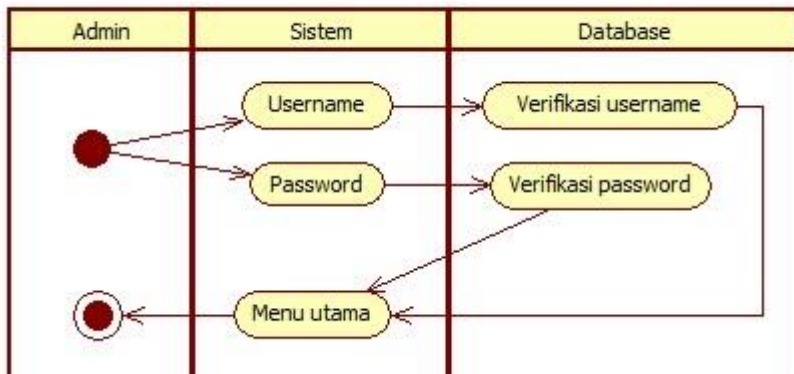
Gambar 3. 27 Communication Diagram View Data Pesanan (UC11)

Pada *communication diagram* ini menjelaskan proses *view data pesanan*. Aktor melakukan *login* terlebih dahulu. Selanjutnya sistem akan memuat form menu utama. Untuk memulai kelola ini, masuk ke menu pesanan, kemudian sistem akan melakukan proses koneksi *database* dan dilanjutkan dengan proses *OpenTable* ke tabel yang dituju yaitu *tbl\_pemesanan* dengan melakukan operasi *select()*. Selanjutnya sistem akan memvalidasi hasil dari operasi, apakah berhasil atau gagal. Dan yang terakhir sistem akan melakukan proses *CloseDB* untuk menutup koneksi.

### 3.2.5 Activity Diagram

*Activity Diagram* menggambarkan berbagai alir aktivitas alam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir [6].

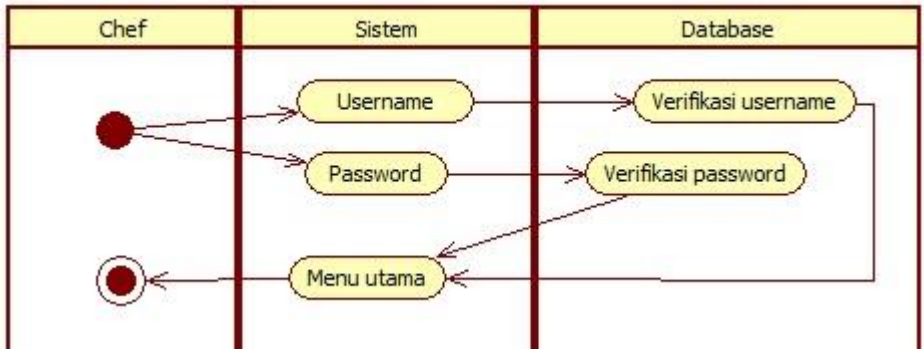
#### 3.2.5.1 Activity Diagram Login Admin (UC1)



Gambar 3. 28 Activity Diagram Login Admin (UC1)

Pada *activity diagram* ini menjelaskan proses *login* admin. Aktor mulai dengan memasukkan *username* dan *password*, kemudian sistem akan memvalidasi apakah *username* dan *password* yang dimasukkan *valid* atau tidak *valid*. Jika *valid* maka akan tampil menu utama, tetapi jika tidak *valid* akan kembali ke *form login*.

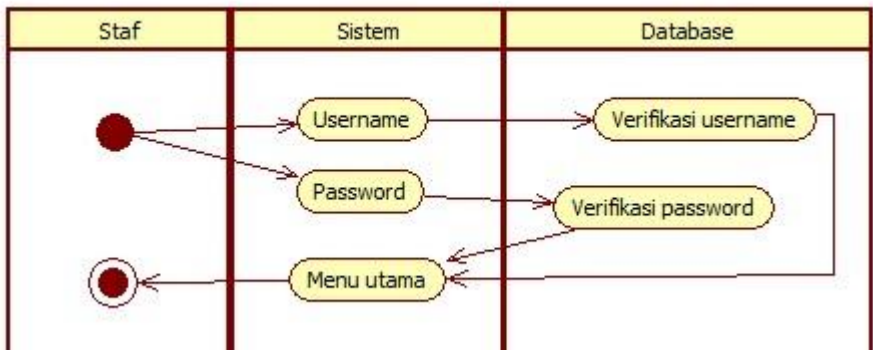
### 3.2.5.2 Activity Diagram Login Chef (UC2)



Gambar 3. 29 Activity Diagram Login Chef (UC2)

Pada *activity diagram* ini menjelaskan proses *login chef*. Aktor mulai dengan memasukkan *username* dan *password*, kemudian sistem akan memvalidasi apakah *username* dan *password* yang dimasukkan *valid* atau tidak *valid*. Jika *valid* maka akan tampil menu utama, tetapi jika tidak *valid* akan kembali ke *form login*.

### 3.2.5.3 Activity Diagram Login Staf (UC3)

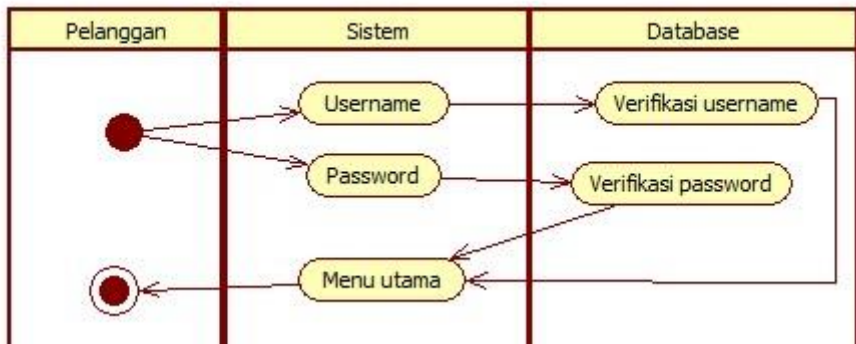


Gambar 3. 30 Activity Diagram Login Staf (UC3)

Pada *activity diagram* ini menjelaskan proses *login staf*.

Aktor mulai dengan memasukkan *username* dan *password*, kemudian sistem akan memvalidasi apakah *username* dan *password* yang dimasukkan *valid* atau tidak *valid*. Jika *valid* maka akan tampil menu utama, tetapi jika tidak *valid* akan kembali ke *form login*.

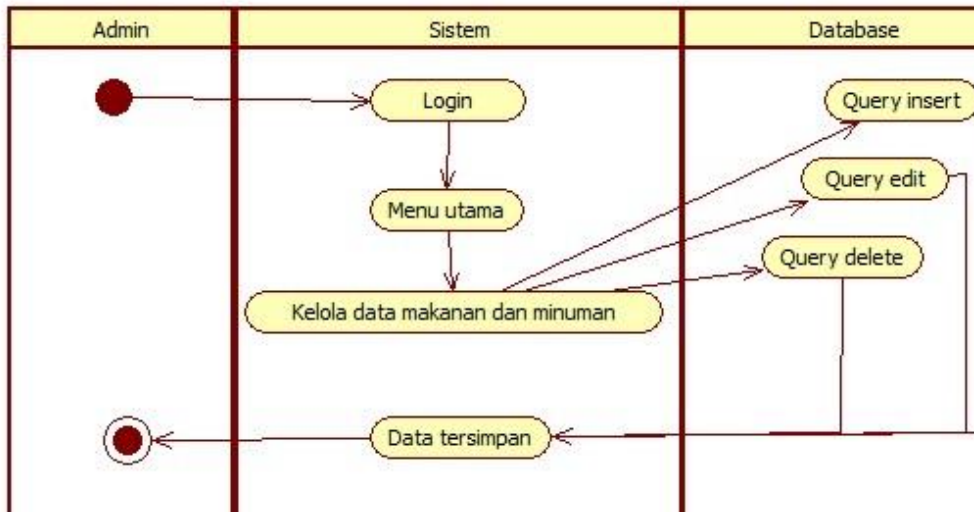
#### 3.2.5.4 Activity Diagram Login Pelanggan (UC4)



Gambar 3. 31 Activity Diagram Login Pelanggan (UC4)

Pada *activity diagram* ini menjelaskan proses *login* pelanggan. Aktor mulai dengan memasukkan *username* dan *password*, kemudian sistem akan memvalidasi apakah *username* dan *password* yang dimasukkan *valid* atau tidak *valid*. Jika *valid* maka akan tampil menu utama, tetapi jika tidak *valid* akan kembali ke *form login*.

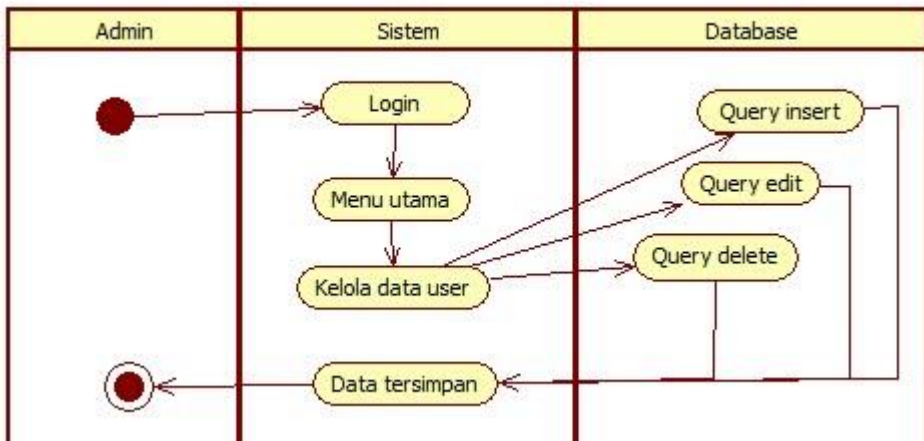
### 3.2.5.5 Activity Diagram Kelola Data Makanan dan Minuman (UC5)



Gambar 3. 32 Activity Diagram Kelola Data Makanan dan Minuman (UC5)

Setelah *login*, Admin memilih menu kelola data makanan dan minuman. Kemudian sistem akan menampilkan kelola data makanan dan minuman. Aktor akan melakukan proses *insert*, *update*, dan *delete* data makanan dan minuman, selanjutnya di sistem akan disimpan data makanan dan minuman. Jika *valid* maka akan tampil data makanan dan minuman, tetapi jika tidak *valid* akan kembali ke kelola data makanan dan minuman. Admin dapat *logout* dan sistem menutup *database*.

### 3.2.5.6 Activity Diagram Kelola Data User (UC6)

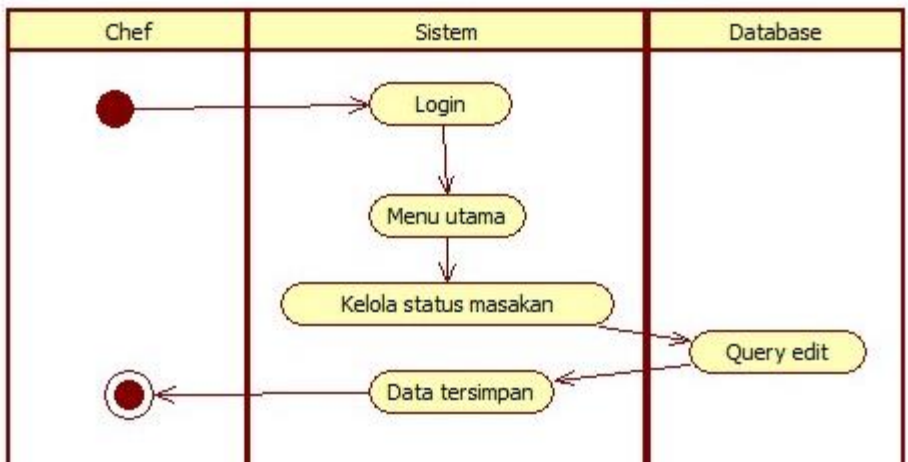


Gambar 3. 33 Activity Diagram Kelola Data User (UC6)

Setelah *login*, Admin memilih menu kelola data *user*. Kemudian sistem akan menampilkan kelola data *user*. Aktor akan melakukan proses *insert*, *update*, dan *delete* data meja, selanjutnya di sistem akan disimpan data *user*. Jika *valid* maka akan tampil data *user*, tetapi jika tidak *valid* akan kembali ke kelola data *user*. Admin dapat *logout* dan sistem menutup *database*.



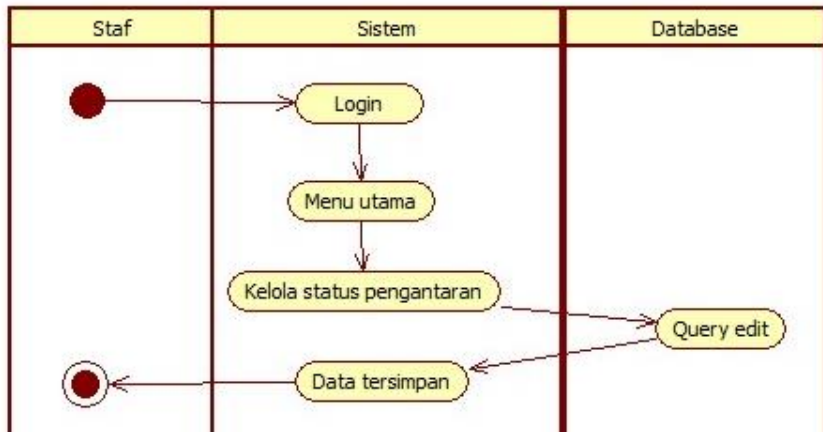
### 3.2.5.7 Activity Diagram Kelola Status Masakan (UC7)



Gambar 3. 34 Activity Diagram Kelola Status Masakan (U7)

Setelah *login*, *Chef* memilih menu kelola status pesanan. Kemudian sistem akan menampilkan kelola status masakan. Aktor akan melakukan proses *update* status pesanan pada bagian masakan, selanjutnya di sistem akan disimpan. Jika *valid* maka akan tampil status masakan, tetapi jika tidak *valid* akan kembali ke kelola status masakan. *Chef* dapat *logout* dan sistem menutup *database*.

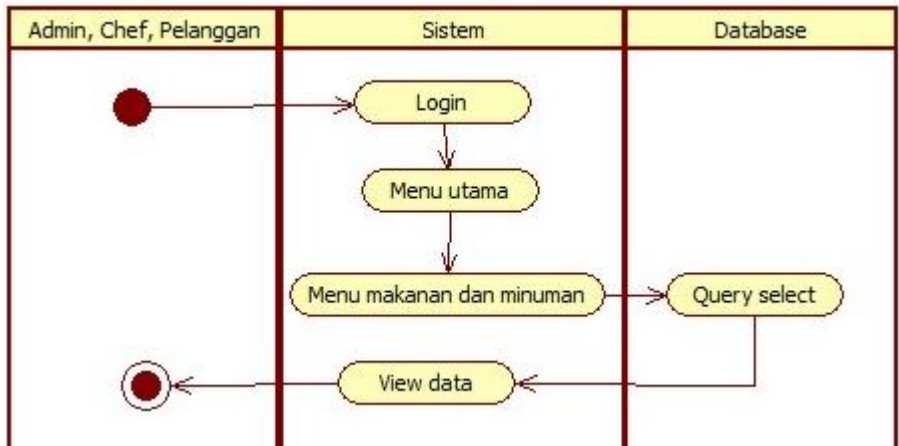
### 3.2.5.8 Activity Diagram Kelola Status Pengantaran (UC8)



Gambar 3. 35 Activity Diagram Kelola Status Pengantaran (UC8)

Setelah *login*, Staf memilih menu kelola status pengantaran. Kemudian sistem akan menampilkan kelola status pengantaran pesanan. Aktor akan melakukan proses *update* status pesanan pada bagian pengantaran pesanan, selanjutnya di sistem akan disimpan. Jika *valid* maka akan tampil status pengantaran pesanan, tetapi jika tidak *valid* akan kembali ke kelola status pengantaran pesanan. Staf dapat *logout* dan sistem menutup *database*.

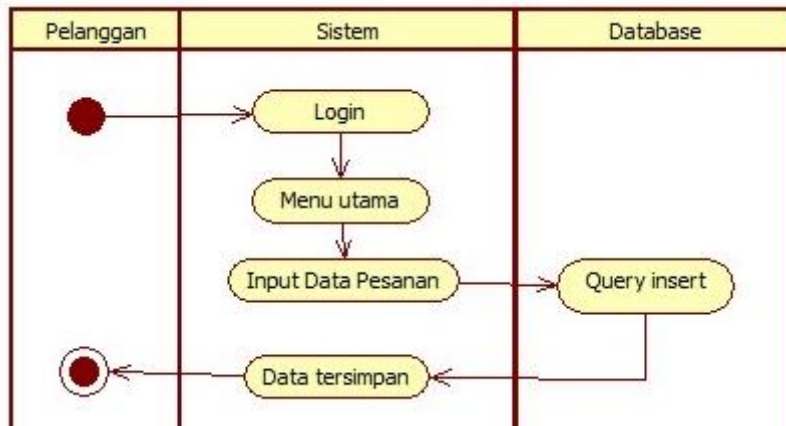
### 3.2.5.9 Activity Diagram View Data Makanan dan Minuman (UC9)



*Gambar 3. 36 Activity Diagram View Data Makanan dan Minuman (UC9)*

Setelah *login*, Admin, Chef, dan Pelanggan memilih menu makanan dan minuman di bagian produk. Kemudian sistem akan menampilkan data makanan dan minuman. Aktor akan melakukan proses *view data* makanan dan minuman, selanjutnya di sistem akan ditampilkan.

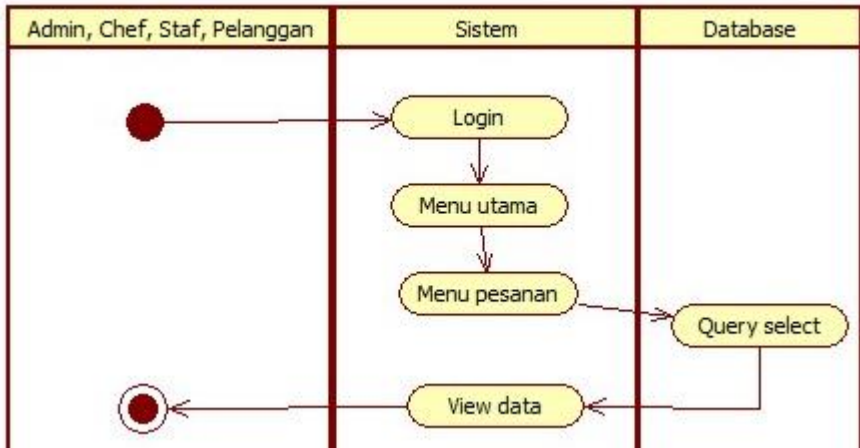
### 3.2.5.10 Activity Diagram Input Data Pesanan (UC10)



Gambar 3. 37 Activity Diagram Input Data Pesanan (UC10)

Setelah *login*, Staf memilih menu kelola status pengantaran. Kemudian sistem akan menampilkan kelola status pengantaran pesanan. Aktor akan melakukan proses *update* status pesanan pada bagian pengantaran pesanan, selanjutnya di sistem akan disimpan. Jika *valid* maka akan tampil status pengantaran pesanan, tetapi jika tidak *valid* akan kembali ke kelola status pengantaran pesanan. Staf dapat *logout* dan sistem menutup *database*.

### 3.2.5.11 Activity Diagram View Data Pesanan (UC11)

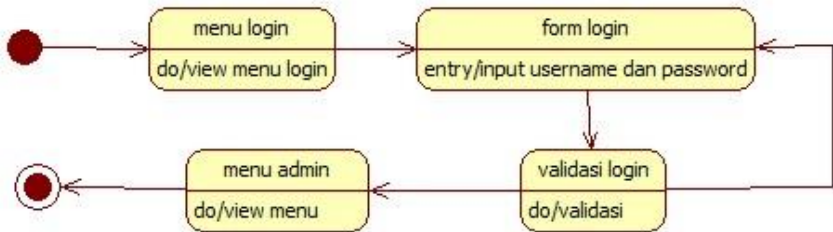


Gambar 3. 38 Activity Diagram View Data Pesanan (UC11)

Setelah *login*, Staf memilih menu kelola status pengantaran. Kemudian sistem akan menampilkan kelola status pengantaran pesanan. Aktor akan melakukan proses *update* status pesanan pada bagian pengantaran pesanan, selanjutnya di sistem akan disimpan. Jika *valid* maka akan tampil status pengantaran pesanan, tetapi jika tidak *valid* akan kembali ke kelola status pengantaran pesanan. Staf dapat *logout* dan sistem menutup *database*.

### 3.2.6 Statechart Diagram

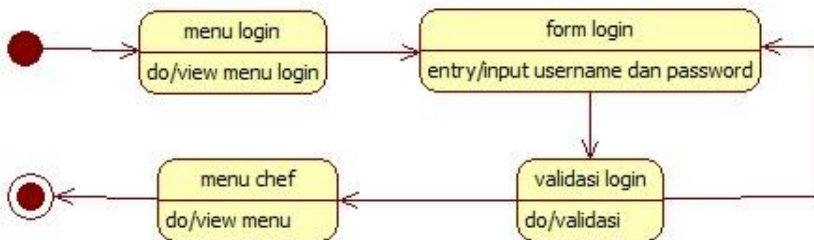
### 3.2.6.1 Statechart Diagram Login Admin (UC1)



Gambar 3. 39 Statechart Diagram Login Admin (UC1)

Admin masuk ke menu *login* menginputkan *username* dan *password*. Jika *valid* maka akan masuk ke menu admin dan jika tidak *valid* akan masuk kembali ke *form login*.

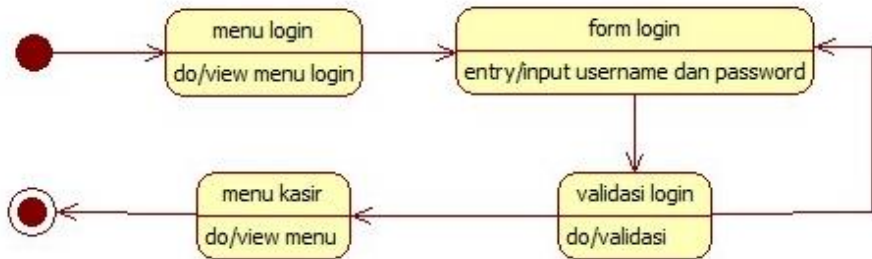
### 3.2.6.2 Statechart Diagram Login Chef (UC2)



Gambar 3. 40 Statechart Diagram Login Chef (UC2)

Chef masuk ke menu *login* menginputkan *username* dan *password*. Jika *valid* maka akan masuk ke menu *chef* dan jika tidak *valid* akan masuk kembali ke *form login*.

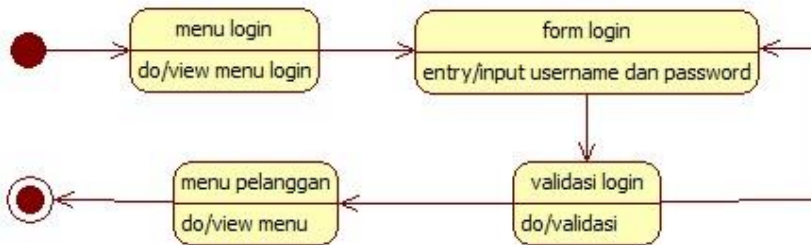
### 3.2.6.3 Statechart Diagram Login Staf (UC3)



Gambar 3. 41 Statechart Diagram Login Staf (UC3)

Staf masuk ke menu *login* menginputkan *username* dan *password*. Jika *valid* maka akan masuk ke menu staf dan jika tidak *valid* akan masuk kembali ke *form login*.

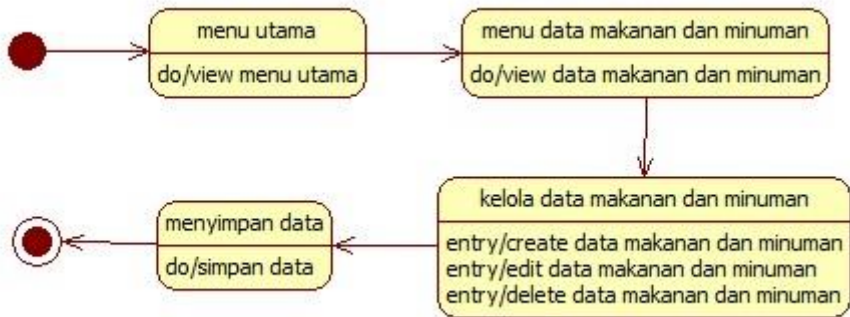
### 3.2.6.4 Statechart Diagram Login Pelanggan (UC4)



Gambar 3. 42 Statechart Diagram Login Pelanggan (UC4)

Pelanggan masuk ke menu *login* menginputkan *username* dan *password* jika *valid* maka akan masuk ke menu pelayan.

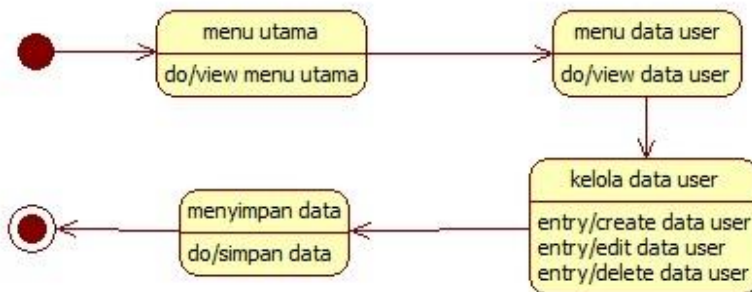
### 3.2.6.5 Statechart Diagram Kelola Data Makanan dan Minuman (UC5)



Gambar 3. 43 Statechart Diagram Kelola Data Makanan dan Minuman (UC5)

Admin mengelola menu data makanan dan minuman yang nanti di validasi ke *database* jika benar maka *query* berhasil.

### 3.2.6.6 Statechart Diagram Kelola Data User (UC6)



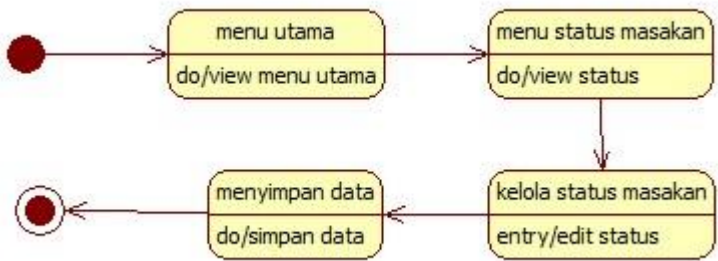
Gambar 3. 44 Statechart Diagram Kelola Data User (UC6)

Admin mengelola menu data makanan dan minuman yang nanti di validasi ke *database* jika benar maka *query*



berhasil.

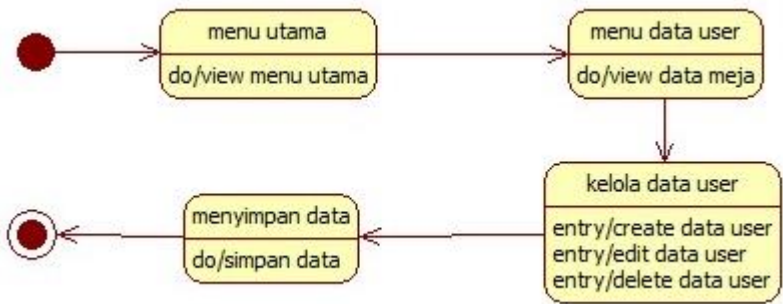
3.2.6.7 Statechart Diagram Kelola Status Masakan (UC7)



Gambar 3. 45 Statechart Diagram Kelola Status Masakan (UC7)

Chef mengelola menu status masakan yang nanti di validasi ke *database* jika benar maka *query* berhasil.

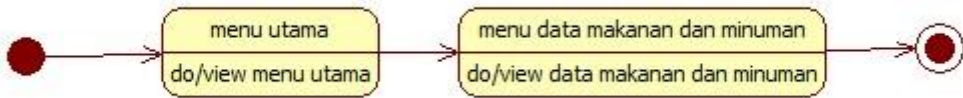
3.2.6.8 Statechart Diagram Kelola Status Pengantaran (UC8)



Gambar 3. 46 Statechart Diagram Kelola Status Pengantaran (UC8)

Staf mengelola menu status pengantaran yang nanti di validasi ke *database* jika benar maka *query* berhasil.

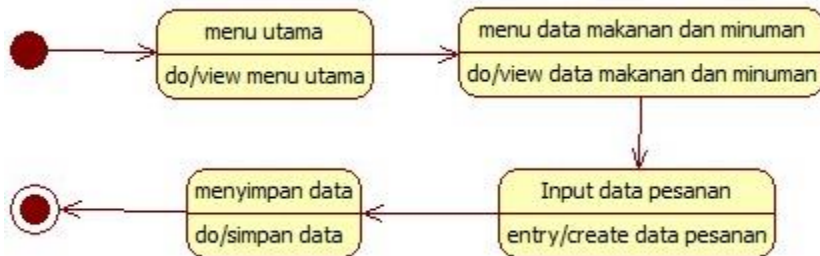
### 3.2.6.9 Statechart Diagram View Data Makanan dan Minuman (UC9)



Gambar 3. 47 Statechart Diagram View Data Makanan dan Minuman (UC9)

Admin, Chef, dan Pelanggan melihat menu makanan dan minuman yang sebelumnya sudah tersimpan di *database*.

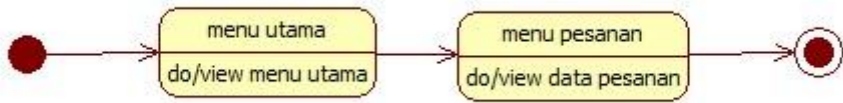
### 3.2.6.10 Statechart Diagram Input Data Pesanan (UC10)



Gambar 3. 48 Statechart Diagram Input Data Pesanan (UC10)

Pelanggan menginputkan data pesanan, kemudian data pesanan akan disimpan di *database*.

### 3.2.6.11 Statechart Diagram View Data Pesanan (UC11)

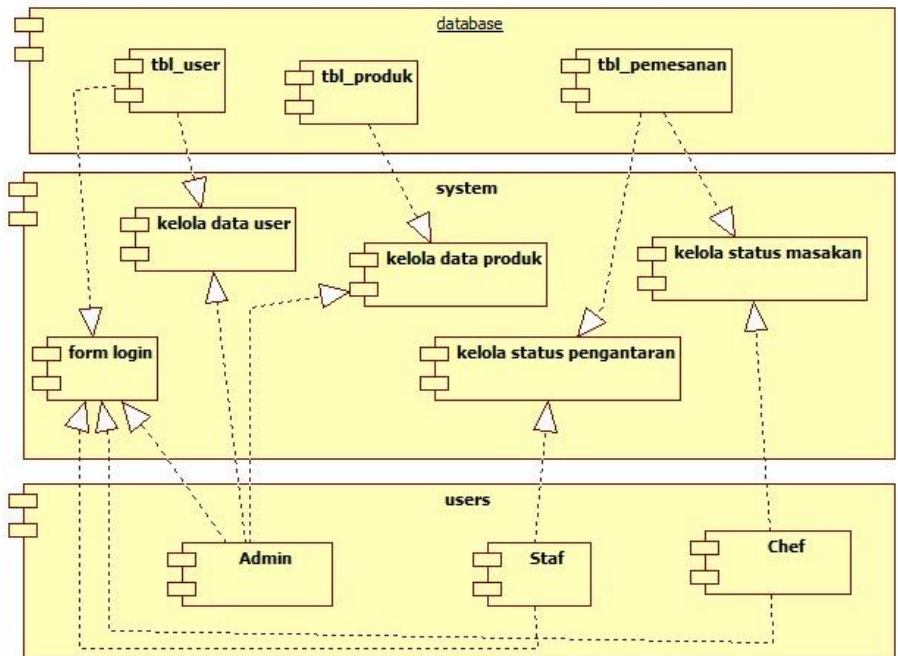


*Gambar 3. 49 Statechart Diagram View Data Pesanan (UC11)*

Admin, *Chef*, Staf dan Pelanggan melihat data pesanan yang sudah tersimpan di *database*.

### 3.2.7 Component Diagram

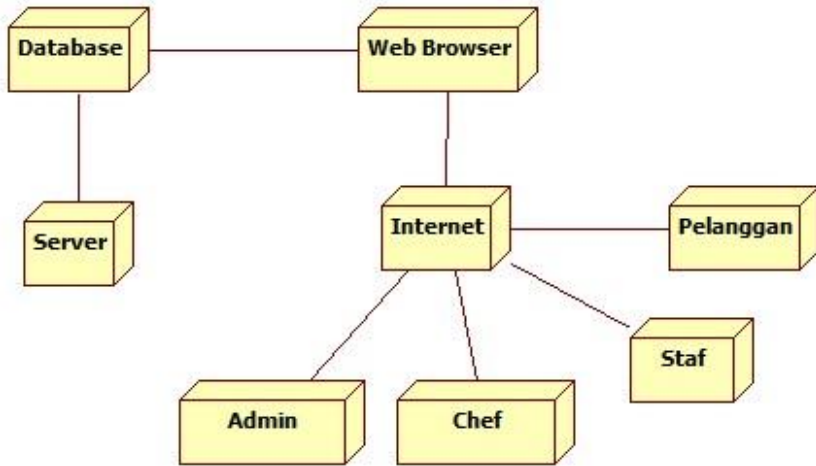
Hal penting pada *component* adalah *component* mewakili potongan-potongan yang independen yang bisa dipesan dan diperbaharui sewaktu-waktu [8]. *Component Diagram* adalah sebuah bentuk pemodelan sebuah sistem kedalam komponen-komponen yang lebih rendah.



Gambar 3. 50 Component Diagram

### 3.2.8 Deployment Diagram

*Deployment diagram* menunjukkan tata letak sebuah sistem secara fisik, menampakkan bagian-bagian *software* yang berjalan pada bagian *hardware*. Bagian utama *hardware* atau perangkat keras adalah node yaitu nama umum untuk semua jenis sumber komputasi [8].



*Gambar 3. 51 Deployment Diagram*

### 3.2.9 Perancangan Antarmuka

Berikut ini adalah perancangan antarmuka aplikasi yang akan dibuat:

#### 3.2.9.1 Halaman *Form Login*

LOGIN

USERNAME

PASSWORD

LOGINDAFTAR

*Gambar 3. 52 Halaman Form Login*

Keterangan :

1. *Login* hanyalah sebuah *label*.
2. *Username* adalah sebuah *textbox*.
3. *Password* adalah sebuah *textbox*.
4. *Login* adalah sebuah *button*.
5. *Daftar* adalah sebuah *button*.

Algoritma :

*If* pilih *button login* *then*

*If* *valid login* *then*

        Masuk kehalaman menu

*Else not valid login*

        Kembali ke halaman *login*

*End if*

*Else if* pilih *button daftar* *then*

    Masuk ke halaman register

*End If*

*End*

### 3.2.9.2 Halaman *Form Register*

REGISTER

NAMA

USERNAME

PASSWORD

ALAMAT

EMAIL

NO. TELP

DAFTAR

*Gambar 3. 53 Halaman Form Register*

Keterangan :

1. *Register* hanyalah sebuah *label*.
2. Nama adalah sebuah *textbox*.
3. *Username* adalah sebuah *textbox*.
4. *Password* adalah sebuah *textbox*.
5. Alamat adalah sebuah *textbox*.
6. *Email* adalah sebuah *textbox*.
7. No. telp adalah sebuah *textbox*.
8. Daftar adalah sebuah *button*.

Algoritma :

*If* pilih *button* daftar *then*

Masuk kehalaman *login*

*End If*

*End*

### 3.2.9.3 Halaman *Home*

FAMILY BRASIL S3

HOME CART LOGIN MENU CONTACT

## FAMILY BRASIL S3

JUMLAH ORANG PILIH TEMPAT PESAN TEMPAT

GAMBAR MENU

Nama Menu

Harga

ADD TO CART

*Gambar 3. 54 Halaman Home*

Keterangan :

1. *Family Brasil S3* hanyalah sebuah *label*.
2. *Home* adalah sebuah *navigasi bar*.
3. *Cart* adalah sebuah *navigasi bar*.
4. *Login* adalah sebuah *navigasi bar*.
5. *Menu* adalah sebuah *navigasi bar*.
6. *Contact* adalah sebuah *navigasi bar*.
7. Jumlah orang adalah sebuah *textbox*.
8. Pilih tempat adalah sebuah *textbox*.



9. Pesan tempat adalah sebuah *button*.
10. Gambar menu adalah sebuah *label*.
11. Nama menu adalah sebuah *label*.
12. *Add to cart* adalah sebuah *button*.

Algoritma :

*Begin if*

Pilih menu : *home*

*Then* menampilkan halaman *home*

*Else if*

Pilih menu : *cart*

*Then* menampilkan halaman *cart*

*Else if*

Pilih menu : *login*

*Then* menampilkan halaman *login*

*Else if*

Pilih menu : *menu*

*Then* menampilkan halaman *menu*

*Else if*

Pilih menu : *contact*

*Then* menampilkan halaman *contact*

*Else if*

Pilih menu : pesan tempat

*Then* menambahkan data berhasil

*Else if*

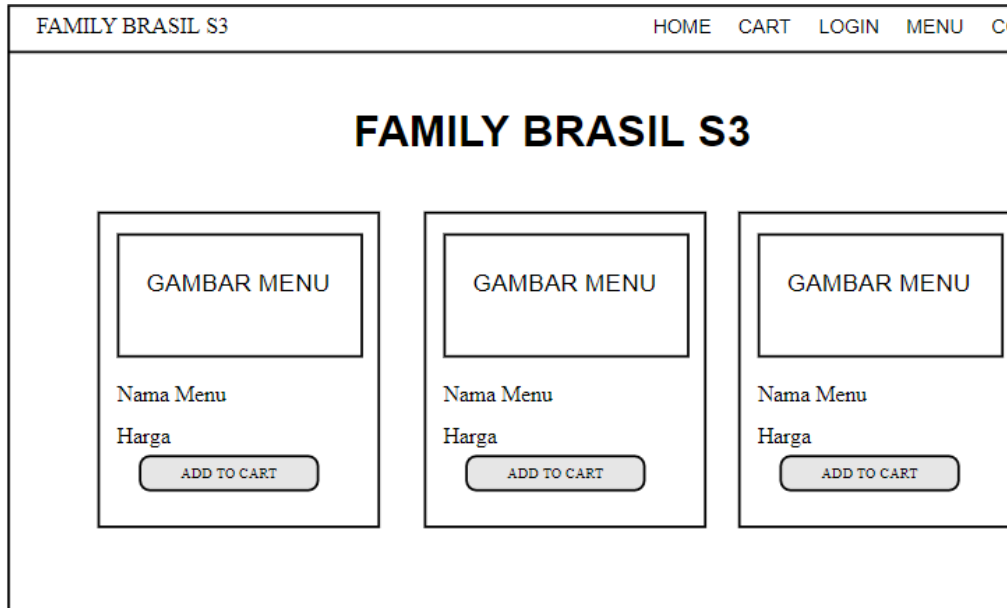
Pilih menu : *add to cart*

*Then* menampilkan halaman *cart*.

*End If*

*End*

### 3.2.9.4 Halaman Pemesanan Menu



*Gambar 3. 55 Halaman Pemesanan Menu*

Keterangan :

1. *Family Brasil S3* hanyalah sebuah *label*.
2. *Home* adalah sebuah *navigasi bar*.
3. *Cart* adalah sebuah *navigasi bar*.
4. *Login* adalah sebuah *navigasi bar*.
5. *Menu* adalah sebuah *navigasi bar*.
6. *Contact* adalah sebuah *navigasi bar*.
7. Gambar menu adalah sebuah *label*.
8. Nama menu adalah sebuah *label*.
9. *Add to cart* adalah sebuah *button*.

Algoritma :

*Begin if*

Pilih menu : *home*

*Then* menampilkan halaman *home*

*Else if*

Pilih menu : *cart*

*Then* menampilkan halaman *cart*

*Else if*

Pilih menu : *login*

*Then* menampilkan halaman *login*

*Else if*

Pilih menu : *menu*

*Then* menampilkan halaman *menu*

*Else if*

Pilih menu : *contact*

*Then* menampilkan halaman *contact*

*Else if*

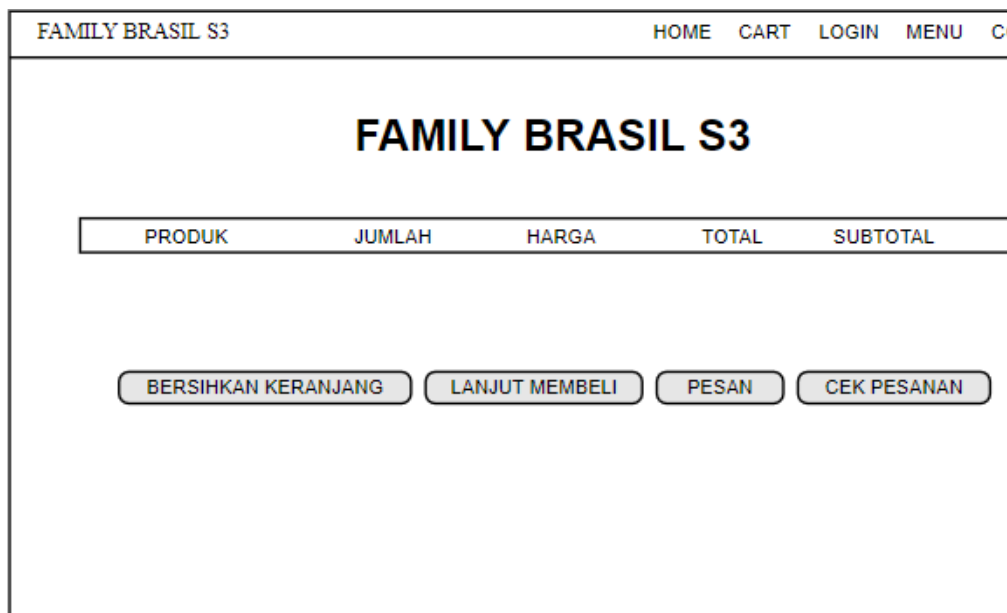
Pilih menu : *add to cart*

*Then* menampilkan halaman *cart*.

*End If*

*End*

### 3.2.9.5 Halaman Keranjang



*Gambar 3. 56 Halaman Keranjang Pesanan*

Keterangan :

1. *Family Brasil S3* hanyalah sebuah *label*.
2. *Home* adalah sebuah *navigasi bar*.
3. *Cart* adalah sebuah *navigasi bar*.
4. *Login* adalah sebuah *navigasi bar*.
5. *Menu* adalah sebuah *navigasi bar*.
6. *Contact* adalah sebuah *navigasi bar*.
7. *Produk* hanyalah sebuah *label*.
8. *Jumlah* hanyalah sebuah *label*.
9. *Harga* hanyalah sebuah *label*.
10. *Total* hanyalah sebuah *label*.
11. *Subtotal* hanyalah sebuah *label*.

12. Bersihkan keranjang adalah sebuah *button*.
13. Lanjut membeli adalah sebuah *button*.
14. Pesan adalah sebuah *button*.
15. Cek pesanan adalah sebuah *button*.

Algoritma :

*Begin if*

Pilih menu : *home*

*Then* menampilkan halaman *home*

*Else if*

Pilih menu : *cart*

*Then* menampilkan halaman *cart*

*Else if*

Pilih menu : *login*

*Then* menampilkan halaman *login*

*Else if*

Pilih menu : *menu*

*Then* menampilkan halaman *menu*

*Else if*

Pilih menu : *contact*

*Then* menampilkan halaman *contact*

*Else if*

Pilih menu : bersihkan keranjang

*Then* menghapus data berhasil

*Else if*

Pilih menu : lanjut membeli

*Then* menampilkan halaman *menu*

*Else if*

Pilih menu : pesan

*Then* menampilkan halaman pembayaran

*Else if*

Pilih menu : cek pesanan

*Then* menampilkan halaman detail pesanan.

*End If*

*End*

### 3.2.9.6 Halaman Pembayaran

FAMILY BRASIL S3 HOME CART LOGIN MENU C

MELAKUKAN PEMBAYARAN KE REKENING DAN MEMBAYAR SECARA CHASH

TANGGAL PEMESANAN	TEMPAT	HARGA	STATUS	DETAIL
-------------------	--------	-------	--------	--------

LANJUT MEMBELI KEMBALI

*Gambar 3. 57 Halaman Pembayaran*

Keterangan :

1. *Family Brasil S3* hanyalah sebuah *label*.
2. *Home* adalah sebuah *navigasi bar*.
3. *Cart* adalah sebuah *navigasi bar*.

4. *Login* adalah sebuah *navigasi bar*.
5. *Menu* adalah sebuah *navigasi bar*.
6. *Contact* adalah sebuah *navigasi bar*.
7. Tanggal pemesanan hanyalah sebuah *label*.
8. Tempat hanyalah sebuah *label*.
9. Harga hanyalah sebuah *label*.
10. Status hanyalah sebuah *label*.
11. Detail hanyalah sebuah *label*.
12. Lanjut membeli adalah sebuah *button*.
13. Kembali adalah sebuah *button*.

Algoritma :

*Begin if*

Pilih menu : *home*

*Then* menampilkan halaman *home*

*Else if*

Pilih menu : *cart*

*Then* menampilkan halaman *cart*

*Else if*

Pilih menu : *login*

*Then* menampilkan halaman *login*

*Else if*

Pilih menu : *menu*

*Then* menampilkan halaman *menu*

*Else if*

Pilih menu : *contact*

*Then* menampilkan halaman *contact*

*Else if*

Pilih menu : lanjut membeli

*Then* menampilkan halaman *menu*

*Else if*

Pilih menu : kembali

*Then* menampilkan halaman *cart*.

*End If*

*End*

### 3.2.9.7 Halaman Admin

	FAMILY BRASIL S3
HOME	SELAMAT DATANG
PRODUK	
USER	
CUSTOMER	
PESANAN	
LAPORAN	
LOGOUT	

*Gambar 3. 58 Halaman Admin*

Keterangan :

1. *Family Brasil S3* hanyalah sebuah *label*.
2. Selamat datang hanyalah sebuah *label*.
3. *Home* adalah sebuah *navigasi bar*.



4. Produk adalah sebuah *navigasi bar*.
5. *User* adalah sebuah *navigasi bar*.
6. *Customer* adalah sebuah *navigasi bar*.
7. Pesanan adalah sebuah *navigasi bar*.
8. Laporan adalah sebuah *navigasi bar*.
9. *Logout* adalah sebuah *navigasi bar*.

Algoritma :

*Begin if*

Pilih menu : *home*

*Then* menampilkan halaman *home*

*Else if*

Pilih menu : produk

*Then* menampilkan halaman produk

*Else if*

Pilih menu : *user*

*Then* menampilkan halaman *user*

*Else if*

Pilih menu : *customer*

*Then* menampilkan halaman *customer*

*Else if*

Pilih menu : pesanan

*Then* menampilkan halaman pesanan

*Else if*

Pilih menu : laporan

*Then* menampilkan halaman laporan

*Else if*

Pilih menu : *logout*

*Then* menampilkan halaman *menu*

*End If*

*End*

3.2.9.8 Halaman Admin Bagian Data Produk

	FAMILY BRASIL S3					
HOME						
PRODUK	ID	FOTO	PRODUK	JENIS	HARGA	ACT
USER						
CUSTOMER						
PESANAN						
LAPORAN						
LOGOUT						

Gambar 3. 59 Halaman Admin Bagian Data Produk

Keterangan :

1. Data produk hanyalah sebuah *label*.
2. Id hanyalah sebuah *label*.
3. Foto hanyalah sebuah *label*.
4. Produk hanyalah sebuah *label*.
5. Jenis hanyalah sebuah *label*.
6. Harga hanyalah sebuah *label*.

7. *Action* hanyalah sebuah *label*.
8. *Home* adalah sebuah *navigasi bar*.
9. *Produk* adalah sebuah *navigasi bar*.
10. *User* adalah sebuah *navigasi bar*.
11. *Customer* adalah sebuah *navigasi bar*.
12. *Pesanan* adalah sebuah *navigasi bar*.
13. *Laporan* adalah sebuah *navigasi bar*.
14. *Logout* adalah sebuah *navigasi bar*.

Algoritma :

*Begin if*

Pilih menu : *home*

*Then* menampilkan halaman *home*

*Else if*

Pilih menu : *produk*

*Then* menampilkan halaman *produk*

*Else if*

Pilih menu : *user*

*Then* menampilkan halaman *user*

*Else if*

Pilih menu : *customer*

*Then* menampilkan halaman *customer*

*Else if*

Pilih menu : *pesanan*

*Then* menampilkan halaman *pesanan*

*Else if*

Pilih menu : *laporan*

*Then* menampilkan halaman laporan

*Else if*

Pilih menu : *logout*

*Then* menampilkan halaman *menu*

*End If*

*End*

### 3.2.9.9 Halaman Admin Bagian Data User

	DATA USER				
HOME					
PRODUK	ID	NAMA USER	USERNAME	LEVEL	ACTION
USER					
CUSTOMER					
PESANAN					
LAPORAN					
LOGOUT					

*Gambar 3. 60 Halaman Admin Bagian Data User*

Keterangan :

1. Data *user* hanyalah sebuah *label*.
2. Id hanyalah sebuah *label*.
3. Nama *user* hanyalah sebuah *label*.
4. *Username* hanyalah sebuah *label*.
5. Level hanyalah sebuah *label*.
6. *Action* hanyalah sebuah *label*.

7. *Home* adalah sebuah *navigasi bar*.
8. *Produk* adalah sebuah *navigasi bar*.
9. *User* adalah sebuah *navigasi bar*.
10. *Customer* adalah sebuah *navigasi bar*.
11. *Pesanan* adalah sebuah *navigasi bar*.
12. *Laporan* adalah sebuah *navigasi bar*.
13. *Logout* adalah sebuah *navigasi bar*.

Algoritma :

*Begin if*

Pilih menu : *home*

*Then* menampilkan halaman *home*

*Else if*

Pilih menu : *produk*

*Then* menampilkan halaman *produk*

*Else if*

Pilih menu : *user*

*Then* menampilkan halaman *user*

*Else if*

Pilih menu : *customer*

*Then* menampilkan halaman *customer*

*Else if*

Pilih menu : *pesanan*

*Then* menampilkan halaman *pesanan*

*Else if*

Pilih menu : *laporan*

*Then* menampilkan halaman *laporan*

*Else if*

Pilih menu : *logout*

*Then* menampilkan halaman *menu*

*End If*

*End*

### 3.2.9.10 Halaman Admin Bagian Data *Customer*

	DATA CUSTOMER					
HOME						
PRODUK	ID	NAMA USER	USERNAME	EMAIL	TELEPON	ACTION
USER						
CUSTOMER						
PESANAN						
LAPORAN						
LOGOUT						

*Gambar 3. 61 Halaman Admin Bagian Data Customer*

Keterangan :

1. Data *customer* hanyalah sebuah *label*.
2. Id hanyalah sebuah *label*.
3. Nama *user* hanyalah sebuah *label*.
4. *Username* hanyalah sebuah *label*.
5. Email hanyalah sebuah *label*.
6. Telepon hanyalah sebuah *label*.
7. *Action* hanyalah sebuah *label*.

8. *Home* adalah sebuah *navigasi bar*.
9. *Produk* adalah sebuah *navigasi bar*.
10. *User* adalah sebuah *navigasi bar*.
11. *Customer* adalah sebuah *navigasi bar*.
12. *Pesanan* adalah sebuah *navigasi bar*.
13. *Laporan* adalah sebuah *navigasi bar*.
14. *Logout* adalah sebuah *navigasi bar*.

Algoritma :

*Begin if*

Pilih menu : *home*

*Then* menampilkan halaman *home*

*Else if*

Pilih menu : *produk*

*Then* menampilkan halaman *produk*

*Else if*

Pilih menu : *user*

*Then* menampilkan halaman *user*

*Else if*

Pilih menu : *customer*

*Then* menampilkan halaman *customer*

*Else if*

Pilih menu : *pesanan*

*Then* menampilkan halaman *pesanan*

*Else if*

Pilih menu : *laporan*

*Then* menampilkan halaman *laporan*

*Else if*

Pilih menu : *logout*

*Then* menampilkan halaman *menu*

*End If*

*End*

### 3.2.9.11 Halaman Admin Bagian Pesanan

	DATA PESANAN						
HOME							
PRODUK	TANGGAL	FAKTUR	NAMA	TEMPAT	EMAIL	TOTAL	ACTION
USER							
CUSTOMER							
PESANAN							
LAPORAN							
LOGOUT							

*Gambar 3. 62 Halaman Admin Bagian Pesanan*

Keterangan :

1. Data pesanan hanyalah sebuah *label*.
2. Tanggal hanyalah sebuah *label*.
3. Faktur hanyalah sebuah *label*.
4. Nama *user* hanyalah sebuah *label*.
5. Tempat hanyalah sebuah *label*.
6. Email hanyalah sebuah *label*.
7. Total hanyalah sebuah *label*.
8. *Action* hanyalah sebuah *label*.



9. *Home* adalah sebuah *navigasi bar*.
10. *Produk* adalah sebuah *navigasi bar*.
11. *User* adalah sebuah *navigasi bar*.
12. *Customer* adalah sebuah *navigasi bar*.
13. *Pesanan* adalah sebuah *navigasi bar*.
14. *Laporan* adalah sebuah *navigasi bar*.
15. *Logout* adalah sebuah *navigasi bar*.

Algoritma :

*Begin if*

Pilih menu : *home*

*Then* menampilkan halaman *home*

*Else if*

Pilih menu : *produk*

*Then* menampilkan halaman *produk*

*Else if*

Pilih menu : *user*

*Then* menampilkan halaman *user*

*Else if*

Pilih menu : *customer*

*Then* menampilkan halaman *customer*

*Else if*

Pilih menu : *pesanan*

*Then* menampilkan halaman *pesanan*

*Else if*

Pilih menu : *laporan*

*Then* menampilkan halaman *laporan*

*Else if*

Pilih menu : *logout*

*Then* menampilkan halaman *menu*

*End If*

*End*

### 3.2.9.12 Halaman Admin Bagian Laporan

	LAPORAN						
HOME	CARI PEMESANAN						
PRODUK	<div>TANGGAL AWAL</div>						
USER	<div>TANGGAL AKHIR</div>						
CUSTOMER							
PESANAN	TANGGAL	FAKTUR	NAMA	TEMPAT	JUMLAH	TOTAL	ACTION
LAPORAN							
LOGOUT							

*Gambar 3. 63 Halaman Admin Laporan*

Keterangan :

1. Data pesanan hanyalah sebuah *label*.
2. Tanggal awal adalah sebuah *date*.
3. Tanggal akhir adalah sebuah *date*.
4. Tanggal hanyalah sebuah *label*.
5. Faktur hanyalah sebuah *label*.
6. Nama *user* hanyalah sebuah *label*.
7. Tempat hanyalah sebuah *label*.

8. Jumlah hanyalah sebuah *label*.
9. Total hanyalah sebuah *label*.
10. *Action* hanyalah sebuah *label*.
11. *Home* adalah sebuah *navigasi bar*.
12. Produk adalah sebuah *navigasi bar*.
13. *User* adalah sebuah *navigasi bar*.
14. *Customer* adalah sebuah *navigasi bar*.
15. Pesanan adalah sebuah *navigasi bar*.
16. Laporan adalah sebuah *navigasi bar*.
17. *Logout* adalah sebuah *navigasi bar*..

Algoritma :

*Begin if*

Pilih menu : *home*

*Then* menampilkan halaman *home*

*Else if*

Pilih menu : produk

*Then* menampilkan halaman produk

*Else if*

Pilih menu : *user*

*Then* menampilkan halaman *user*

*Else if*

Pilih menu : *customer*

*Then* menampilkan halaman *customer*

*Else if*

Pilih menu : pesanan

*Then* menampilkan halaman pesanan

*Else if*

Pilih menu : laporan

*Then* menampilkan halaman laporan

*Else if*

Pilih menu : *logout*

*Then* menampilkan halaman *menu*

*End If*

*End*

### 3.2.9.13 Halaman *Chef*

	FAMILY BRASIL S3
HOME	SELAMAT DATANG
PRODUK	
USER	
CUSTOMER	
PESANAN	
LAPORAN	
LOGOUT	

*Gambar 3. 64 Halaman Chef*

Keterangan :

1. *Family Brasil S3* hanyalah sebuah *label*
2. Selamat datang hanyalah sebuah *label*

3. *Home* adalah sebuah *navigasi bar*.
4. *Produk* adalah sebuah *navigasi bar*.
5. *Customer* adalah sebuah *navigasi bar*.
6. *Pesanan* adalah sebuah *navigasi bar*.
7. *Laporan* adalah sebuah *navigasi bar*.
8. *Logout* adalah sebuah *navigasi bar*.

Algoritma :

*Begin if*

Pilih menu : *home*

*Then* menampilkan halaman *home*

*Else if*

Pilih menu : *produk*

*Then* menampilkan halaman *produk*

*Else if*

Pilih menu : *customer*

*Then* menampilkan halaman *customer*

*Else if*

Pilih menu : *pesanan*

*Then* menampilkan halaman *pesanan*

*Else if*

Pilih menu : *laporan*

*Then* menampilkan halaman *laporan*

*Else if*

Pilih menu : *logout*

*Then* menampilkan halaman *menu*

*End If*

*End*

### 3.2.9.14 Halaman *Chef* Bagian Produk

	DATA PRODUK				
HOME					
PRODUK	ID	FOTO	PRODUK	JENIS	HARGA
CUSTOMER					
PESANAN					
LAPORAN					
LOGOUT					

*Gambar 3. 65 Halaman Chef Bagian Produk*

Keterangan :

1. Data produk hanyalah sebuah *label*.
2. Id hanyalah sebuah *label*.
3. Foto hanyalah sebuah *label*.
4. Produk hanyalah sebuah *label*.
5. Jenis hanyalah sebuah *label*.
6. Harga hanyalah sebuah *label*.
7. *Home* adalah sebuah *navigasi bar*.
8. Produk adalah sebuah *navigasi bar*.
9. *Customer* adalah sebuah *navigasi bar*.
10. Pesanan adalah sebuah *navigasi bar*.
11. Laporan adalah sebuah *navigasi bar*.

12. *Logout* adalah sebuah *navigasi bar*.

Algoritma :

*Begin if*

Pilih menu : *home*

*Then* menampilkan halaman *home*

*Else if*

Pilih menu : produk

*Then* menampilkan halaman produk

*Else if*

Pilih menu : *customer*

*Then* menampilkan halaman *customer*

*Else if*

Pilih menu : pesanan

*Then* menampilkan halaman pesanan

*Else if*

Pilih menu : laporan

*Then* menampilkan halaman laporan

*Else if*

Pilih menu : *logout*

*Then* menampilkan halaman *menu*

*End If*

*End*

### 3.2.9.15 Halaman *Chef* Bagian *Customer*

	DATA CUSTOMER				
HOME					
PRODUK	ID	NAMA USER	USERNAME	EMAIL	TELEPON
CUSTOMER					
PESANAN					
LAPORAN					
LOGOUT					

*Gambar 3. 66 Halaman Chef Bagian Customer*

Keterangan :

1. Data customer hanyalah sebuah *label*.
2. Id hanyalah sebuah *label*.
3. Nama *user* hanyalah sebuah *label*.
4. *Username* hanyalah sebuah *label*.
5. Email hanyalah sebuah *label*.
6. Telepon hanyalah sebuah *label*.
7. *Home* adalah sebuah *navigasi bar*.
8. Produk adalah sebuah *navigasi bar*.
9. *Customer* adalah sebuah *navigasi bar*.
10. Pesanan adalah sebuah *navigasi bar*.
11. Laporan adalah sebuah *navigasi bar*.
12. *Logout* adalah sebuah *navigasi bar*.



Algoritma :

*Begin if*

Pilih menu : *home*

*Then* menampilkan halaman *home*

*Else if*

Pilih menu : produk

*Then* menampilkan halaman produk

*Else if*

Pilih menu : *customer*

*Then* menampilkan halaman *customer*

*Else if*

Pilih menu : pesanan

*Then* menampilkan halaman pesanan

*Else if*

Pilih menu : laporan

*Then* menampilkan halaman laporan

*Else if*

Pilih menu : *logout*

*Then* menampilkan halaman *menu*

*End If*

*End*

3.2.9.16 Halaman *Chef* Bagian Pesanan

	DATA PESANAN						
HOME							
PRODUK	TANGGAL	FAKTUR	NAMA	TEMPAT	JUMLAH	TOTAL	ACTION
CUSTOMER							
PESANAN							
LAPORAN							
LOGOUT							

Gambar 3. 67 Halaman *Chef* Bagian Pesanan

Keterangan :

1. Data pesanan hanyalah sebuah *label*.
2. Tanggal hanyalah sebuah *label*.
3. Faktur hanyalah sebuah *label*.
4. Nama hanyalah sebuah *label*.
5. Tempat hanyalah sebuah *label*.
6. Jumlah hanyalah sebuah *label*.
7. Total hanyalah sebuah *label*.
8. *Action* hanyalah sebuah *label*.
9. *Home* adalah sebuah *navigasi bar*.
10. *Produk* adalah sebuah *navigasi bar*.
11. *Customer* adalah sebuah *navigasi bar*.
12. *Pesanan* adalah sebuah *navigasi bar*.
13. *Laporan* adalah sebuah *navigasi bar*.

14. *Logout* adalah sebuah *navigasi bar*.

Algoritma :

*Begin if*

Pilih menu : *home*

*Then* menampilkan halaman *home*

*Else if*

Pilih menu : produk

*Then* menampilkan halaman produk

*Else if*

Pilih menu : *customer*

*Then* menampilkan halaman *customer*

*Else if*

Pilih menu : pesanan

*Then* menampilkan halaman pesanan

*Else if*

Pilih menu : laporan

*Then* menampilkan halaman laporan

*Else if*

Pilih menu : *logout*

*Then* menampilkan halaman *menu*

*End If*

*End*

3.2.9.17 Halaman *Chef* Bagian Laporan

	LAPORAN							
HOME	<div>CARI PEMESANAN</div> <div><div>TANGGAL AWAL</div><div>TANGGAL AKHIR</div></div>							
PRODUK								
CUSTOMER								
PESANAN								
LAPORAN	TANGGAL	FAKTUR	NAMA	TEMPAT	JUMLAH	TOTAL	ACTION	
LOGOUT								

Gambar 3. 68 Halaman *Chef* Bagian Laporan

Keterangan :

- 1. Data pesanan hanyalah sebuah *label*.
- 2. Tanggal awal adalah sebuah *date*.
- 3. Tanggal akhir adalah sebuah *date*.
- 4. Tanggal hanyalah sebuah *label*.
- 5. Faktur hanyalah sebuah *label*.
- 6. Nama hanyalah sebuah *label*.
- 7. Tempat hanyalah sebuah *label*.
- 8. Jumlah hanyalah sebuah *label*.
- 9. Total hanyalah seb
- 10. Action hanyalah sebuah *label*.
- 11. *Home* adalah sebuah *navigasi bar*.
- 12. *Produk* adalah sebuah *navigasi bar*.
- 13. *Customer* adalah sebuah *navigasi bar*.

14. Pesanan adalah sebuah *navigasi bar*.
15. Laporan adalah sebuah *navigasi bar*.
16. *Logout* adalah sebuah *navigasi bar*.

Algoritma :

*Begin if*

Pilih menu : *home*

*Then* menampilkan halaman *home*

*Else if*

Pilih menu : produk

*Then* menampilkan halaman produk

*Else if*

Pilih menu : *customer*

*Then* menampilkan halaman *customer*

*Else if*

Pilih menu : pesanan

*Then* menampilkan halaman pesanan

*Else if*

Pilih menu : laporan

*Then* menampilkan halaman laporan

*Else if*

Pilih menu : *logout*

*Then* menampilkan halaman *menu*

*End If*

*End*

### 3.2.9.18 Halaman Staf

	FAMILY BRASIL S3
HOME	SELAMAT DATANG
PRODUK	
USER	
CUSTOMER	
PESANAN	
LAPORAN	
LOGOUT	

*Gambar 3. 69 Halaman Staf*

Keterangan :

1. *Family Brasil S* hanyalah sebuah *label*.
2. Selamat datang hanyalah sebuah *label*.
3. *Home* adalah sebuah *navigasi bar*.
4. *Produk* adalah sebuah *navigasi bar*.
5. *Customer* adalah sebuah *navigasi bar*.
6. *Pesanan* adalah sebuah *navigasi bar*.
7. *Laporan* adalah sebuah *navigasi bar*.
8. *Logout* adalah sebuah *navigasi bar*.

Algoritma :

*Begin if*

Pilih menu : *home*

*Then* menampilkan halaman *home*

*Else if*

Pilih menu : produk

*Then* menampilkan halaman produk

*Else if*

Pilih menu : *customer*

*Then* menampilkan halaman *customer*

*Else if*

Pilih menu : pesanan

*Then* menampilkan halaman pesanan

*Else if*

Pilih menu : laporan

*Then* menampilkan halaman laporan

*Else if*

Pilih menu : *logout*

*Then* menampilkan halaman *menu*

*End If*

*End*

3.2.9.19 Halaman Staf Bagian Produk

	DATA PRODUK				
HOME					
PRODUK	ID	FOTO	PRODUK	JENIS	HARGA
CUSTOMER					
PESANAN					
LAPORAN					
LOGOUT					

Gambar 3. 70 Halaman Staf Bagian Produk

Keterangan :

- 1. Data produk hanyalah sebuah *label*.
- 2. Id hanyalah sebuah *label*.
- 3. Foto hanyalah sebuah *label*.
- 4. Produk hanyalah sebuah *label*.
- 5. Jenis hanyalah sebuah *label*.
- 6. Harga hanyalah sebuah *label*.
- 7. *Home* adalah sebuah *navigasi bar*.
- 8. Produk adalah sebuah *navigasi bar*.
- 9. *Customer* adalah sebuah *navigasi bar*.
- 10. Pesanan adalah sebuah *navigasi bar*.
- 11. Laporan adalah sebuah *navigasi bar*.
- 12. *Logout* adalah sebuah *navigasi bar*.



Algoritma :

*Begin if*

Pilih menu : *home*

*Then* menampilkan halaman *home*

*Else if*

Pilih menu : produk

*Then* menampilkan halaman produk

*Else if*

Pilih menu : *customer*

*Then* menampilkan halaman *customer*

*Else if*

Pilih menu : pesanan

*Then* menampilkan halaman pesanan

*Else if*

Pilih menu : laporan

*Then* menampilkan halaman laporan

*Else if*

Pilih menu : *logout*

*Then* menampilkan halaman *menu*

*End If*

*End*

3.2.9.20 Halaman Staf Bagian *Customer*

	DATA CUSTOMER				
HOME					
PRODUK	ID	NAMA USER	USERNAME	EMAIL	TELEPON
CUSTOMER					
PESANAN					
LAPORAN					
LOGOUT					

*Gambar 3. 71 Halaman Staf Bagian Customer*

Keterangan :

1. Data customer hanyalah sebuah *label*.
2. Id hanyalah sebuah *label*.
3. Nama *user* hanyalah sebuah *label*.
4. *Username* hanyalah sebuah *label*.
5. Email hanyalah sebuah *label*.
6. Telepon hanyalah sebuah *label*.
7. *Home* adalah sebuah *navigasi bar*.
8. Produk adalah sebuah *navigasi bar*.
9. *Customer* adalah sebuah *navigasi bar*.
10. Pesanan adalah sebuah *navigasi bar*.
11. Laporan adalah sebuah *navigasi bar*.
12. *Logout* adalah sebuah *navigasi bar*.

Algoritma :

*Begin if*

Pilih menu : *home*

*Then* menampilkan halaman *home*

*Else if*

Pilih menu : produk

*Then* menampilkan halaman produk

*Else if*

Pilih menu : *customer*

*Then* menampilkan halaman *customer*

*Else if*

Pilih menu : pesanan

*Then* menampilkan halaman pesanan

*Else if*

Pilih menu : laporan

*Then* menampilkan halaman laporan

*Else if*

Pilih menu : *logout*

*Then* menampilkan halaman *menu*

*End If*

*End*

3.2.9.21 Halaman Staf Bagian Pesanan

	DATA PESANAN					
HOME						
PRODUK	TANGGAL	FAKTUR	NAMA	TEMPAT	TOTAL	ACTION
CUSTOMER						
PESANAN						
LAPORAN						
LOGOUT						

Gambar 3. 72 Halaman Staf Bagian Pesanan

Keterangan :

1. Data pesanan hanyalah sebuah *label*.
2. Tanggal hanyalah sebuah *label*.
3. Faktur hanyalah sebuah *label*.
4. Nama hanyalah sebuah *label*.
5. Tempat hanyalah sebuah *label*.
6. Total hanyalah sebuah *label*.
7. *Action* hanyalah sebuah *label*.
8. *Home* adalah sebuah *navigasi bar*.
9. *Produk* adalah sebuah *navigasi bar*.
10. *Customer* adalah sebuah *navigasi bar*.
11. *Pesanan* adalah sebuah *navigasi bar*.
12. *Laporan* adalah sebuah *navigasi bar*.
13. *Logout* adalah sebuah *navigasi bar*.

Algoritma :

*Begin if*

Pilih menu : *home*

*Then* menampilkan halaman *home*

*Else if*

Pilih menu : produk

*Then* menampilkan halaman produk

*Else if*

Pilih menu : *customer*

*Then* menampilkan halaman *customer*

*Else if*

Pilih menu : pesanan

*Then* menampilkan halaman pesanan

*Else if*

Pilih menu : laporan

*Then* menampilkan halaman laporan

*Else if*

Pilih menu : *logout*

*Then* menampilkan halaman *menu*

*End If*

*End*

3.2.9.22 Halaman Staf Bagian Laporan

	LAPORAN							
HOME	CARI PEMESANAN							
PRODUK	<div>TANGGAL AWAL</div>							
CUSTOMER	<div>TANGGAL AKHIR</div>							
PESANAN								
LAPORAN	TANGGAL	FAKTUR	NAMA	TEMPAT	JUMLAH	TOTAL	ACTION	
LOGOUT								

Gambar 3. 73 Halaman Staf Bagian Laporan

Keterangan :

- 1. Data pesanan hanyalah sebuah *label*.
- 2. Tanggal awal adalah sebuah *date*.
- 3. Tanggal akhir adalah sebuah *date*.
- 4. Tanggal hanyalah sebuah *label*.
- 5. Faktur hanyalah sebuah *label*.
- 6. Nama hanyalah sebuah *label*.
- 7. Tempat hanyalah sebuah *label*.
- 8. Jumlah hanyalah sebuah *label*.
- 9. Total hanyalah sebuah *label*.
- 10. *Action* hanyalah sebuah *label*.
- 11. *Home* adalah sebuah *navigasi bar*.
- 12. *Produk* adalah sebuah *navigasi bar*.
- 13. *Customer* adalah sebuah *navigasi bar*.

14. Pesanan adalah sebuah *navigasi bar*.
15. Laporan adalah sebuah *navigasi bar*.
16. *Logout* adalah sebuah *navigasi bar*..

Algoritma :

*Begin if*

Pilih menu : *home*

*Then* menampilkan halaman *home*

*Else if*

Pilih menu : produk

*Then* menampilkan halaman produk

*Else if*

Pilih menu : *customer*

*Then* menampilkan halaman *customer*

*Else if*

Pilih menu : pesanan

*Then* menampilkan halaman pesanan

*Else if*

Pilih menu : laporan

*Then* menampilkan halaman laporan

*Else if*

Pilih menu : *logout*

*Then* menampilkan halaman *menu*

*End If*

*End*

## BAB 3

---

# CARA INSTALASI TOOLS

---

### 3.1 Code Igniter

CodeIgniter merupakan sebuah framework PHP yang sederhana serta mudah untuk dipelajari. Selain itu CodeIgniter merupakan framework PHP yang menggunakan konsep desain MVC yaitu Model, View, Controller.



### **3.1.1 Cara Instalasi Code Ignitor**

#### **▪ Buka web Codeigniter**

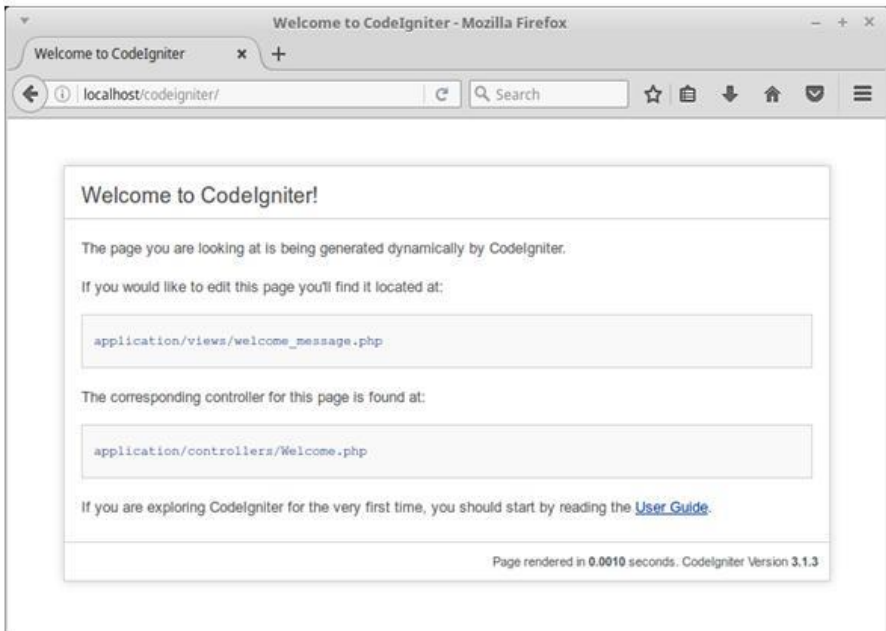
Aplikasi Manajemen Proyek Politeknik Pos Indonesia.

By Luthfi M. Nabil, Hagan Rowlenstino A. S Copyright c 2020 Kreatif Industri Nusantara.



Gambar 3.1 Web Code Ignitor

- Lalu Download Codeigniter.
- Kemudian simpan Codeigniter pada file htdocs di xampp
- Selanjutnya lakukan ekstrak pada file tersebut.
- Aktifkan Xamppnya, lalu buka browser dan ketikan `http://localhost/codeigniter`
- Maka akan muncul tampilan seperti dibawah. CodeIgniter telah berhasil melakukan instalasi.



Gambar 3.2 Hasil

## 3.2 XAMPP

Xampp merupakan sebuah perangkat lunak bebas, yang mendukung banyak sistem operasi serta kompilasi dari beberapa program. Selain itu xampp terdiri dari satu paket aplikasi lengkap yang dibutuhkan untuk membuah sebuah web server, serta di dalamnya terdapat Apache, Mysql,dan Php.

### 3.2.1 Kelebihan XAMPP

1. Database storage engine banyak digunakan oleh programmer web developer dikarenakan sifatnya yang bebas.
2. Xampp memiliki kapasitas yan cukup sekitar 60.000 tabel dengan record men-capai sekitar 5.000.000.000 bahkan xampp yang terbaru sudah lebih jumlah recordnya.
3. Pada xampp terdapat keamanan datanya cukup aman meskipun tidak sebanding dengan postgre dan oracle.
4. Engine pada xampp ini multiplatform sehingga mampu diimplementasikan di berbagai system operasi
5. Xampp mempunyai kelebihan utama yaitu kecepatannya.

### 3.2.2 Kekurangan XAMPP

1. Xampp tidak cocok dalam menangani data dengan jumlah yang sangat be-sar, baik untuk menyimpan data ataupun untuk memproses data.
2. Serta memiliki keterbatasan pada kemampuan kinerja pada server ketika suatu data disimpan telah melebihi batas maksimal, maka kemampuan daya tamping server tidak dapat menerapkan konsep Technology Cluste.

### 3.2.3 Cara Instalasi XAMPP di Windows

1. Download terlebih dahulu xampp nya, sesuai dengan windows yang dimiliki.
2. Setelah itu simpan data xampp di data C, lalu double klik pada mentahan xampp, seperti gambar dibawah.
3. Kemudian akan muncul splash screen bitnami, setelah itu akan ada pesan eror, itu terjadi karena masalah permission atau hak akses, lalu klik ok karena akan dikonfigurasi ke hard drive lain.
4. Setelah itu akan muncul bagian intaslasi dari Setup Wizard Xampp, seperti gam-bar dibawah ini.



Gambar 3.3 XAMPP

5. Kemudian akan muncul halaman pemilihan komponen apa saja yang ingin instal pada web server xampp ini, lalu masuk dan sesuaikan modul-modul instalasi sesuai kebutuhan. Jika sudah sesuai, klik Next saja.