

SKRIPSI

PENGEMBANGAN SISTEM INFORMASI KEUANGAN DAN INVENTORI BERBASIS WEB PADA RESTORAN TA CHIA

Ditulis untuk memenuhi sebagian persyaratan akademik
guna memperoleh gelar Sarjana Komputer

Oleh:

NAMA : FILBERT CHAN

NPM : 01081190011



**PROGRAM STUDI SISTEM INFORMASI
FAKULTAS ILMU KOMPUTER
UNIVERSITAS PELITA HARAPAN
JAKARTA
2023**



UNIVERSITAS PELITA HARAPAN

PERNYATAAN DAN PERSETUJUAN UNGGAH TUGAS AKHIR

Saya/kami yang bertanda tangan dibawah ini:

Nama – NPM : Filbert Chan - 01081190011

Fakultas : Ilmu Komputer

Program Studi : Sistem Informasi

Lokasi Kampus : Tangerang

Jenis Tugas Akhir : Skripsi

Judul :

“PENGEMBANGAN SISTEM INFORMASI KEUANGAN DAN INVENTORIS BERBASIS WEB PADA RESTORAN TA CHIA”

Menyatakan bahwa:

1. Tugas akhir tersebut adalah benar karya saya/kami dengan arahan dari dosen pembimbing dan bukan merupakan duplikasi karya tulis yang sudah dipublikasikan atau yang pernah dipakai untuk mendapatkan gelar akademik di perguruan tinggi manapun.
2. Tugas akhir tersebut bukan merupakan plagiat dari hasil karya pihak lain, dan apabila saya/kami mengutip dari karya orang lain maka akan dicantumkan sebagai referensi sesuai dengan ketentuan yang berlaku.
3. Saya/kami memberikan Hak Noneksklusif Tanpa Royalti kepada Universitas Pelita Harapan atas Tugas Akhir tersebut untuk diunggah ke dalam Repositori UPH.

Apabila di kemudian hari ditemukan pelanggaran Hak Cipta dan Kekayaan Intelektual atau Peraturan Perundangan-undangan Republik Indonesia lainnya dan integritas akademik dalam karya saya/kami tersebut, maka saya/kami bersedia menanggung secara pribadi segala bentuk tuntutan hukum dan sanksi akademis yang timbul serta membebaskan Universitas Pelita Harapan dari segala tuntutan hukum yang berlaku.

Dibuat di : Tangerang

Pada Tanggal : 2 Januari 2023

Yang menyatakan,

Tanda Tangan	
Nama	(Filbert Chan)



**UNIVERSITAS PELITA HARAPAN
FAKULTAS ILMU KOMPUTER**

**PERSETUJUAN DOSEN PEMBIMBING TUGAS AKHIR
PENGEMBANGAN SISTEM INFORMASI KEUANGAN DAN INVENTORIS
BERBASIS WEB PADA RESTORAN TA CHIA**

Oleh :


Nama : Filbert Chan
NPM : 01081190011
Program Studi : Sistem Informasi
Peminatan : *Web and Mobile Development*

Telah diperiksa dan disetujui untuk diajukan dan dipertahankan dalam Sidang Tugas Akhir guna memperoleh gelar Strata Satu Sarjana Komputer (S.Kom.) pada Program Studi Sistem Informasi, Fakultas Ilmu Komputer, Universitas Pelita Harapan, Tangerang, Banten.

Tangerang, 28 Desember 2022

Menyetujui :

Pembimbing Pertama


(Kusno Prasetya, Ph.D.)

Pembimbing Kedua


(Calandra Alencia Haryani, S.E., S.SI.,
M.T.I.)

**Ketua Program Studi
Sistem Informasi**


(Arnold Aribowo, S.T., M.T.)

Dekan Fakultas Ilmu Komputer


(Dr. Eng. Ir. Pujianto Yugopuspito, MSc.)








UNIVERSITAS PELITA HARAPAN
FAKULTAS ILMU KOMPUTER

PERSETUJUAN TIM PENGUJI TUGAS AKHIR

Pada hari Jumat, 13 Januari 2023 telah diselenggarakan Sidang Tugas Akhir mahasiswa/i dengan keterangan:

Nama : Filbert Chan
NPM : 01081190011
Judul : PENGEMBANGAN SISTEM INFORMASI KEUANGAN DAN INVENTORIS BERBASIS WEB PADA RESTORAN TA CHIA

Sebagai persyaratan akademik untuk mencapai gelar Strata Satu Sarjana Komputer (S.Kom.) pada Program Studi Sistem Informasi, Fakultas Ilmu Komputer, Universitas Pelita Harapan, Tangerang, Banten, yang diuji oleh tim penguji yang terdiri dari:

Nama	Status	Tanda tangan
Kusno Prasetya, Ph.D.	sebagai ketua sidang	
Aditya Rama Mitra, S.Si., M.T.	sebagai anggota	 31 Jan 2023
Arnold Aribowo, S.T., M.T.	sebagai anggota	

ABSTRAK

Filbert Chan (01081190011)

PENGEMBANGAN SISTEM INFORMASI KEUANGAN DAN INVENTORI BERBASIS WEB PADA RESTORAN TA CHIA

(xix+268 halaman: 241 gambar, 114 tabel, 5 lampiran)

Proses pencatatan keuangan dan inventori pada setiap perusahaan besar telah menggunakan teknologi untuk membantu efektivitas dan efisiensi kerja. Restoran Ta Chia merupakan salah satu usaha yang bergerak pada bidang kuliner yang menyediakan masakan *Chinese Food* yang non-halal. Sistem pekerjaan dalam Restoran Ta Chia yang mencakup pencatatan dan pemesanan pada saat ini masih dilakukan secara manual dengan menggunakan pencatatan pada kertas dan buku. Hal ini dapat mengakibatkan beberapa masalah yang timbul seperti data yang tertulis secara tidak akurat atau kehilangan data.

Oleh karena itu, tujuan dari tugas akhir ini adalah untuk merancang dan membangun sebuah sistem informasi pencatatan keuangan dan inventori berbasis web pada Restoran Ta Chia untuk membuat data-data yang tercatat dapat tersimpan dan tercatat dengan mudah, sehingga meningkatkan efisiensi dan efektivitas proses bisnis. Sistem ini dikembangkan menggunakan metodologi pengembangan sistem *prototyping* yang dilakukan menggunakan bahasa pemrograman PHP dengan *framework* CodeIgniter. Pemodelan sistem dibuat menggunakan notasi UML, yakni *use case diagram*, *activity diagram*, *class diagram*, dan *entity relationship diagram*.

Hasil dari tugas akhir yang telah dilakukan adalah aplikasi berbasis *web* yang dapat digunakan untuk pencatatan dan pengelolaan data-data keuangan, inventori, dan lain-lainnya, serta membantu mempermudah proses pemesanan dengan menghubungkan pihak kasir dengan dapur/bartender.

Kata kunci: PHP, CodeIgniter, keuangan, inventori, sistem informasi

Referensi: 20 (2003 – 2022)

ABSTRACT

Filbert Chan (01081190011)

DEVELOPMENT OF WEB-BASED FINANCIAL AND INVENTORY INFORMATION SYSTEM AT RESTAURANT TA CHIA

(xix + 268 pages: 241 figures, 114 tables, 5 appendices)

The recording process of financial and inventory at every large company has implemented technology in their work to improve effectiveness and efficiency. Ta Chia is a restaurant business that is engaged in the culinary field that provides non-halal Chinese food. The work system at Ta Chia, which includes recording data and ordering dishes, are currently still being done manually using pens and papers. This can result in several problems arising such as an inaccurately written data or data loss.

Therefore, the purpose of this final project is to design and build a web-based inventory and financial recording information system at Ta Chia to make recorded data stored and recorded easily, thereby increasing the efficiency and effectiveness of the restaurant's work. This system was developed using a prototyping system development methodology that was carried out using the PHP programming language with the CodeIgniter framework. The modeling system is being made using UML notation, namely use case diagrams, activity diagrams, class diagrams, and entity relationship diagrams.

The result of the final project that has been carried out is a web-based application that can be used for recording and managing financial data, inventory, and others, as well as to help simplify the ordering process communication by connecting the cashier with the kitchen/bartender.

Keywords: PHP, CodeIgniter, finance, inventory, information systems

References: 20 (2003 – 2022)

KATA PENGANTAR

Puji dan syukur kepada Tuhan Yang Maha Esa atas segala anugerah yang diberikan-Nya sehingga penulis dapat menyelesaikan Tugas Akhir yang berjudul “PENGEMBANGAN SISTEM INFORMASI KEUANGAN DAN INVENTORI BERBASIS WEB PADA RESTORAN TA CHIA”.

Tugas Akhir ini disusun untuk memenuhi sebagian dari persyaratan akademik guna memperoleh gelar Sarjana Komputer Strata Satu Program Studi Sistem Informasi Universitas Pelita Harapan, Karawaci, Tangerang.

Penulis menyadari bahwa tanpa bimbingan, bantuan, dan doa, serta dukungan dari berbagai pihak, Tugas Akhir ini tidak dapat diselesaikan tepat pada waktunya. Oleh karena itu, penulis mengucapkan terima kasih yang sebesar-besarnya kepada semua pihak yang telah membantu proses pengerjaan Tugas Akhir ini, yaitu kepada:

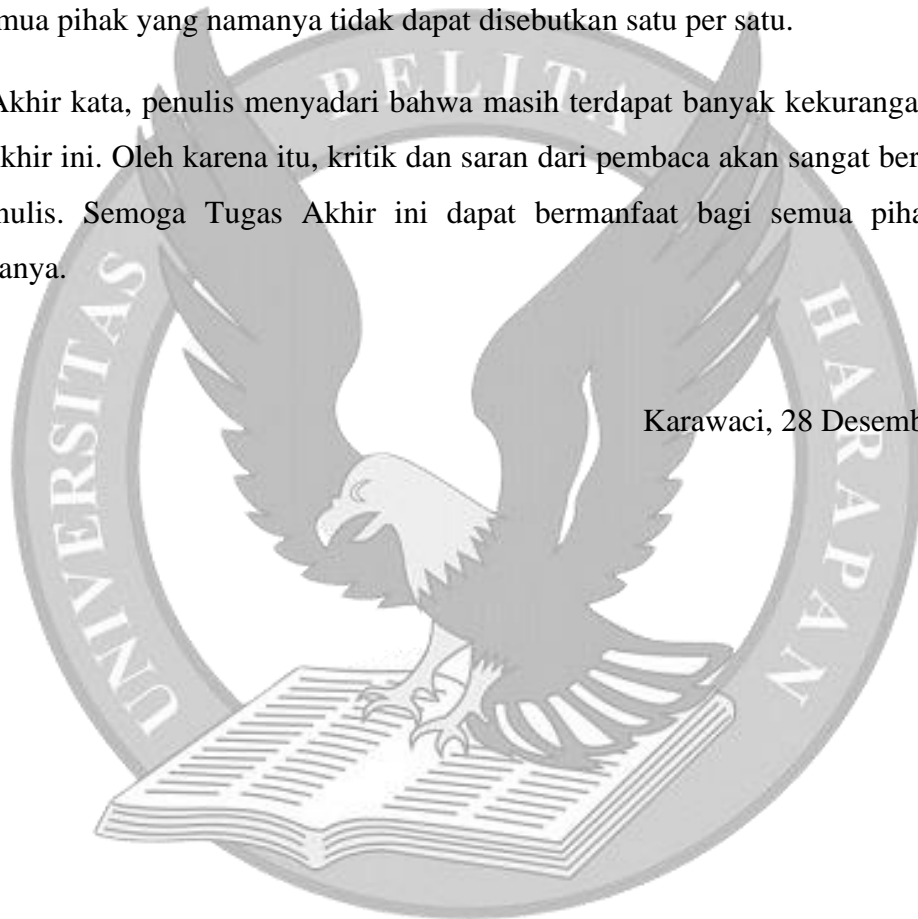
- 1) Bapak Dr. Eng. Ir. Pujiyanto Yugopuspito, M.Sc., selaku Dekan Fakultas Ilmu Komputer.
- 2) Bapak Arnold Aribowo, S.T., M.T., selaku Ketua Program Studi Sistem Informasi Universitas Pelita Harapan dan Penasehat Akademik penulis.
- 3) Ibu Calandra Alencia Haryani, S.E., S.SI., M.T.I., selaku Dosen Pembimbing Utama penulis yang telah meluangkan waktu, serta memberikan arahan, dukungan, dan motivasi kepada penulis, baik selama proses perkuliahan maupun selama proses pengerjaan Tugas Akhir.
- 4) Bapak Kusno Prasetya, Ph.D., selaku Dosen Pembimbing Pendamping penulis yang telah meluangkan waktu, serta memberikan arahan, dukungan, dan motivasi kepada penulis, baik selama proses perkuliahan maupun selama proses pengerjaan Tugas Akhir.
- 5) Seluruh dosen yang telah memberikan dukungan, ajaran, dan nasihat kepada penulis selama penulis menempuh pendidikan di Program Studi Sistem Informasi, Universitas Pelita Harapan.
- 6) Seluruh staf Fakultas Ilmu Komputer Universitas Pelita Harapan yang telah membantu penulis dalam kegiatan administratif.
- 7) Manajer Restoran Ta Chia yang telah meluangkan waktu, menyediakan informasi, dan memberikan masukan kepada penulis selama pengerjaan Tugas Akhir.

- 8) Orang tua, adik, dan keluarga besar penulis yang memberikan dukungan kepada penulis sehingga dapat menyelesaikan Tugas Akhir ini.
- 9) Teman-teman seperjuangan Program Studi Sistem Informasi Angkatan 2019, yang telah berjuang bersama, menemani dan memberikan semangat kepada penulis selama masa perkuliahan.
- 10) Ko Josiah, Ci Valeri, dan Ci Valen, selaku mentor yang setia membimbing dan memberikan masukan-masukan kepada penulis selama masa perkuliahan.
- 11) Teman-teman seperjuangan antar jurusan dan antar angkatan yang telah menemani dan mendukung penulis selama masa perkuliahan.
- 12) Semua pihak yang namanya tidak dapat disebutkan satu per satu.

Akhir kata, penulis menyadari bahwa masih terdapat banyak kekurangan dalam Tugas Akhir ini. Oleh karena itu, kritik dan saran dari pembaca akan sangat bermanfaat bagi penulis. Semoga Tugas Akhir ini dapat bermanfaat bagi semua pihak yang membacanya.

Karawaci, 28 Desember 2022

Penulis



DAFTAR ISI

ABSTRAK.....	i
ABSTRACT.....	vi
KATA PENGANTAR	vii
DAFTAR ISI.....	ix
DAFTAR GAMBAR	xii
DAFTAR TABEL.....	xvii
BAB I PENDAHULUAN	1
1.1. Latar Belakang.....	1
1.2. Rumusan Masalah	2
1.3. Batasan Masalah.....	2
1.4. Tujuan dan Manfaat Penelitian.....	2
1.4.1. Tujuan Penelitian	2
1.4.2. Manfaat Penelitian	2
1.5. Metodologi Penelitian.....	3
1.5.1. Metodologi Pengumpulan Data.....	3
1.5.2. Metode Pengembangan Sistem	3
1.6. Sistematika Penulisan	4
BAB II LANDASAN TEORI	5
2.1. Kajian Teori.....	5
2.1.1. Restoran	5
2.1.2. Komputerisasi	6
2.1.3. <i>Inventory</i>	6
2.1.4. Sistem Informasi Keuangan	7
2.1.5. <i>Web Programming</i>	7
2.1.5.1. PHP.....	7
2.1.5.2. Codeigniter	8
2.1.5.3. HTML.....	9
2.1.5.4. CSS	9
2.1.5.5. Javascript	9
2.1.6. Sistem Basis Data (<i>Database</i>).....	10
2.1.7. <i>Entity Relationship Diagram (ERD)</i>	11
2.1.8. <i>System Development Life Cycle (SDLC)</i>	13
2.1.8.1. Tahapan Perencanaan (<i>Planning</i>)	13
2.1.8.2. Tahapan Analisis (<i>Analysis</i>)	14
2.1.8.3. Tahapan Desain (<i>Design</i>).....	14
2.1.8.4. Tahapan Implementasi (<i>Implementation</i>)	14
2.1.9. Metode <i>Prototype</i>	15
2.1.10. <i>Unified Modelling Language (UML)</i>	16
2.1.10.1. Pemodelan Struktural.....	16

2.1.10.1.1.	<i>Class Diagram</i>	16
2.1.10.2.	Pemodelan Perilaku	17
2.1.10.2.1.	<i>Use Case Diagram</i>	18
2.1.10.2.2.	<i>Activity Diagram</i>	19
2.1.11.	Pengujian UML.....	21
2.2.	Penelitian Terdahulu.....	22
BAB III SISTEM SAAT INI		23
3.1.	Profil Perusahaan.....	23
3.2.	Struktur Organisasi	24
3.3.	Analisis Sistem Saat Ini.....	25
3.3.1.	Kendala Sistem Saat Ini	34
BAB IV ANALISIS DAN PERANCANGAN SISTEM USULAN		35
4.1.	Analisis Kelayakan Sistem	35
4.1.1.	Kelayakan Teknis (<i>Technical Feasibility</i>).....	35
4.1.2.	Kelayakan Organisasi (<i>Organizational Feasibility</i>).....	35
4.1.3.	Kelayakan Ekonomi (<i>Economic Feasibility</i>).....	36
4.2.	Tahap Analisis	37
4.2.1.	Pengumpulan Kebutuhan Pengguna (<i>User Requirement Gathering</i>).....	37
4.2.1.1.	<i>Functional Requirement</i>	38
4.2.1.2.	<i>Non-Functional Requirement</i>	40
4.3.	Sistem Usulan	40
4.3.1.	Pemodelan Fungsional Pemodelan Struktural.....	40
4.3.1.1.	<i>Use Case Diagram</i>	41
4.3.1.2.	<i>Activity Diagram</i>	42
4.3.2.	Pemodelan Struktural	66
4.3.3.	Perancangan Manajemen Data.....	68
4.3.3.1.	<i>Entity Relationship Diagram</i>	68
4.3.3.2.	<i>Table Description</i>	69
4.4.	Implementasi Sistem.....	75
4.4.1.	<i>Mockup</i>	75
4.4.2.	Perancangan Lapisan Antarmuka (<i>User Interface</i>)	98
4.4.2.1.	Perancangan Lapisan Antarmuka (<i>Landing Page</i>)	98
4.4.2.2.	Perancangan Lapisan Antarmuka (Pihak Manajer).....	102
4.4.2.3.	Perancangan Lapisan Antarmuka (Pihak Kasir)	140
4.4.2.4.	Perancangan Lapisan Antarmuka (Pihak Dapur).....	147
4.4.3.	Tahap Pengujian.....	149
BAB V KESIMPULAN DAN SARAN		152
5.1	Kesimpulan.....	152
5.2	Saran	153
DAFTAR PUSTAKA		154
LAMPIRAN A		A-1

LAMPIRAN B B-1
LAMPIRAN C C-1
LAMPIRAN D D-1
LAMPIRAN E E-1



DAFTAR GAMBAR

Gambar 2.1.Entity ERD	12
Gambar 2.2 Attribute ERD	12
Gambar 2.3 Proses model prototyping	16
Gambar 3.1 Struktur Organisasi Usaha Restoran Ta Chia	24
Gambar 3.3 Kertas Order Restoran Ta Chia	26
Gambar 3.2 <i>Activity Diagram</i> Restoran Ta Chia (Pemesanan Pelanggan)	27
Gambar 3.5 Bon Pembayaran Restoran Ta Chia.....	28
Gambar 3.4 <i>Activity Diagram</i> Restoran Ta Chia (Proses Pembayaran).....	29
Gambar 3.6 <i>Activity Diagram</i> Restoran Ta Chia (Pembuatan Laporan Laba Rugi)	31
Gambar 3.7 <i>Activity Diagram</i> Restoran Ta Chia (Pengecekan Inventori)	33
Gambar 4.1 <i>Use Case Diagram</i> Sistem Usulan	41
Gambar 4.2 <i>Activity Diagram Sign Up</i>	42
Gambar 4.3 <i>Activity Diagram</i> Mengelola <i>User</i>	43
Gambar 4.4 <i>Activity Diagram</i> Mengelola <i>Role</i>	44
Gambar 4.5 <i>Activity Diagram</i> Mengelola <i>Menu</i>	45
Gambar 4.6 <i>Activity Diagram</i> Mengelola <i>Detail Menu</i>	46
Gambar 4.7 <i>Activity Diagram</i> Mengelola <i>Kategori Menu</i>	47
Gambar 4.8 <i>Activity Diagram</i> Mengelola <i>Inventori Barang</i>	48
Gambar 4.9 <i>Activity Diagram</i> Manambah <i>Stok Inventori Barang</i>	49
Gambar 4.10 <i>Activity Diagram</i> Mengelola <i>Jenis Barang</i>	50
Gambar 4.11 <i>Activity Diagram</i> Mengelola <i>Satuan Barang</i>	51
Gambar 4.12 <i>Activity Diagram</i> Mengelola <i>Pemasukan</i>	52
Gambar 4.13 <i>Activity Diagram</i> Mengelola <i>Meja</i>	53
Gambar 4.14 <i>Activity Diagram</i> Mengelola <i>Metode Bayar</i>	54
Gambar 4.15 <i>Activity Diagram</i> Mengelola <i>Pelanggan</i>	55
Gambar 4.16 <i>Activity Diagram</i> Mengelola <i>Pengeluaran</i>	56
Gambar 4.17 <i>Activity Diagram</i> Mengelola <i>Supplier</i>	57
Gambar 4.18 <i>Activity Diagram</i> Mengelola <i>Pesanan Pelanggan</i>	58
Gambar 4.19 <i>Activity Diagram</i> Mengelola <i>Detail Pesanan</i>	60
Gambar 4.20 <i>Activity Diagram</i> Melakukan <i>Konfirmasi Pesanan</i>	62
Gambar 4.21 <i>Activity Diagram</i> Memberikan <i>Diskon Pesanan</i>	63
Gambar 4.22 <i>Activity Diagram</i> Bayar <i>Pesanan</i>	65
Gambar 4.23 <i>Class Diagram</i> Sistem Usulan	67
Gambar 4.24 <i>Entity Relationship Diagram</i> Sistem Usulan.....	68
Gambar 4.25 Rancangan Halaman <i>Log In</i>	75
Gambar 4.26 Rancangan Halaman <i>Sign Up</i>	76
Gambar 4.27 Rancangan Halaman <i>Dashboard</i> (Manajer).....	76
Gambar 4.28 Rancangan Halaman <i>Dashboard</i> (Kasir).....	77
Gambar 4.29 Rancangan Halaman <i>Dashboard</i> (Dapur)	77
Gambar 4.30 Rancangan Halaman <i>Master Data User</i>	78
Gambar 4.31 Rancangan Halaman Menambah <i>Data User</i>	78
Gambar 4.32 Rancangan Halaman Mengubah <i>Data User</i>	79
Gambar 4.33 Rancangan Halaman <i>Master Data Role</i>	79
Gambar 4.34 Rancangan Halaman Menambah <i>Data Role</i>	80
Gambar 4.35 Rancangan Halaman Mengubah <i>Data Role</i>	80
Gambar 4.36 Perancangan Halaman <i>Master Data Menu</i>	80
Gambar 4.37 Perancangan Halaman Menambah <i>Data Menu</i>	81
Gambar 4.38 Perancangan Halaman Mengubah <i>Data Menu</i>	81
Gambar 4.39 Perancangan Halaman <i>Data Detail Menu</i>	82
Gambar 4.40 Perancangan Halaman Menambah <i>Data Detail Menu</i>	82
Gambar 4.41 Perancangan Halaman Mengubah <i>Data Detail Menu</i>	83
Gambar 4.42 Perancangan Halaman <i>Master Data Kategori</i>	83
Gambar 4.43 Perancangan Halaman Menambah <i>Data Kategori</i>	84
Gambar 4.44 Perancangan Halaman Mengubah <i>Data Kategori</i>	84
Gambar 4.45 Perancangan Halaman <i>Master Data Barang</i>	84

Gambar 4.46 Perancangan Halaman Menambah Data Barang	85
Gambar 4.47 Perancangan Halaman Mengubah Data Barang	85
Gambar 4.48 Perancangan Halaman Menambah Jumlah Stok Barang	86
Gambar 4.49 Perancangan Halaman <i>Master Data</i> Jenis Barang.....	86
Gambar 4.50 Perancangan Halaman Menambah Data Jenis Barang	87
Gambar 4.51 Perancangan Halaman Mengubah Data Jenis Barang	87
Gambar 4. 52 Perancangan Halaman <i>Master Data</i> Satuan	87
Gambar 4.53 Perancangan Halaman Menambah Data Satuan.....	88
Gambar 4.54 Perancangan Halaman Mengubah Data Satuan.....	88
Gambar 4.55 Perancangan Halaman <i>Master Data</i> Meja.....	88
Gambar 4.56 Perancangan Halaman Menambah Data Meja.....	89
Gambar 4.57 Perancangan Halaman Mengubah Data Meja	89
Gambar 4.58 Perancangan Halaman <i>Master Data</i> Metode Bayar	90
Gambar 4.59 Perancangan Halaman Menambah Data Metode Bayar	90
Gambar 4.60 Perancangan Halaman Mengubah Data Metode Bayar	90
Gambar 4.61 Perancangan Halaman <i>Master Data</i> Pelanggan	91
Gambar 4.62 Perancangan Halaman Menambah Data Pelanggan	91
Gambar 4.63 Perancangan Halaman Mengubah Data Pelanggan	92
Gambar 4.64 Perancangan Halaman <i>Master Data</i> Pengeluaran	92
Gambar 4.65 Perancangan Halaman Menambah Data Pengeluaran	93
Gambar 4.66 Perancangan Halaman Mengubah Data Pengeluaran	93
Gambar 4.67 Perancangan Halaman <i>Master Data Supplier</i>	94
Gambar 4.68 Perancangan Halaman Menambah Data <i>Supplier</i>	94
Gambar 4.69 Perancangan Halaman Mengubah Data <i>Supplier</i>	95
Gambar 4.70 Perancangan Halaman <i>Master Data</i> Pesanan	95
Gambar 4.71 Perancangan Halaman Menambah Data Pesanan.....	95
Gambar 4.72 Perancangan Halaman Data <i>Detail</i> Pesanan (Manajer).....	96
Gambar 4.73 Perancangan Halaman Data <i>Detail</i> Pesanan (Kasir)	96
Gambar 4.74 Perancangan Halaman Menambah Data Pesanan <i>Detail</i>	97
Gambar 4.75 Perancangan Halaman Menambah Diskon Pesanan.....	97
Gambar 4.76 Perancangan Halaman Bayar Pesanan.....	98
Gambar 4.77 Perancangan Halaman <i>Bill</i> Pesanan	98
Gambar 4.78 Tampilan Halaman <i>Log In</i>	99
Gambar 4.79 Tampilan notifikasi data <i>input</i> salah atau kolom kosong pada halaman <i>Log In</i>	99
Gambar 4.80 Tampilan notifikasi <i>user input</i> salah <i>password</i> pada halaman <i>Log In</i>	100
Gambar 4.81 Tampilan notifikasi <i>user</i> tidak dapat mengakses pada halaman <i>Log In</i>	100
Gambar 4.82 Tampilan Halaman <i>Sign Up</i>	101
Gambar 4.83 Tampilan notifikasi kolom kosong pada halaman <i>Sign Up</i>	101
Gambar 4.84 Tampilan notifikasi data sudah ada pada halaman <i>Sign Up</i>	102
Gambar 4.85 Tampilan Halaman <i>Dashboard</i> Manajer (1)	102
Gambar 4.86 Tampilan Halaman <i>Dashboard</i> Manajer (2)	103
Gambar 4.87 Tampilan notifikasi konfirmasi <i>log out</i>	103
Gambar 4.88 Tampilan Halaman <i>Master Data User</i>	104
Gambar 4.89 Tampilan Halaman Menambah Data <i>User</i>	104
Gambar 4.90 Tampilan Halaman Mengubah Data <i>User</i>	105
Gambar 4.91 Tampilan notifikasi data sudah ada pada halaman <i>master data user</i>	105
Gambar 4.92 Tampilan notifikasi konfirmasi hapus data <i>user</i>	106
Gambar 4.93 Tampilan PDF <i>Master Data User</i>	106
Gambar 4.94 Tampilan Halaman <i>Master Data Role</i>	107
Gambar 4.95 Tampilan Halaman Menambah Data <i>Role</i>	107
Gambar 4.96 Tampilan Halaman Mengubah Data <i>Role</i>	108
Gambar 4.97 Tampilan notifikasi data sudah ada pada halaman <i>master data role</i>	108
Gambar 4.98 Tampilan notifikasi konfirmasi hapus data <i>role</i>	108
Gambar 4.99 Tampilan PDF <i>Master Data Role</i>	109
Gambar 4.100 Tampilan Halaman <i>Master Data Menu</i>	109
Gambar 4.101 Tampilan Halaman Menambah Data <i>Menu</i>	110
Gambar 4.102 Tampilan Halaman Mengubah Data <i>Menu</i>	110

Gambar 4.103 Tampilan notifikasi kode sudah ada pada halaman <i>master data menu</i>	111
Gambar 4.104 Tampilan notifikasi data sudah ada pada halaman <i>master data menu</i>	111
Gambar 4.105 Tampilan status data <i>menu inactive</i>	111
Gambar 4.106 Tampilan PDF <i>Master Data Menu</i>	112
Gambar 4.107 Tampilan Halaman <i>Master Data Detail Menu</i>	112
Gambar 4.108 Tampilan Halaman Menambah Data <i>Detail Menu</i>	113
Gambar 4.109 Tampilan Halaman <i>Edit Data Detail Menu</i>	113
Gambar 4.110 Tampilan notifikasi data sudah ada pada halaman <i>master data detail menu</i>	114
Gambar 4.111 Tampilan notifikasi konfirmasi hapus data <i>detail menu</i>	114
Gambar 4.112 Tampilan Halaman <i>Master Data Kategori</i>	114
Gambar 4.113 Tampilan Halaman Menambah Data Kategori.....	115
Gambar 4.114 Tampilan Halaman Mengubah Data Kategori.....	115
Gambar 4.115 Tampilan notifikasi data sudah ada pada halaman <i>master data kategori</i>	116
Gambar 4.116 Tampilan notifikasi konfirmasi hapus data kategori.....	116
Gambar 4.117 Tampilan PDF <i>Master Data Kategori</i>	116
Gambar 4.118 Tampilan Halaman <i>Master Data Barang</i>	117
Gambar 4.119 Tampilan Halaman Menambah Data Barang.....	117
Gambar 4.120 Tampilan Halaman Mengubah Data Barang.....	118
Gambar 4.121 Tampilan Halaman Tambah Stok Barang.....	119
Gambar 4.122 Tampilan notifikasi data sudah ada pada halaman <i>master data barang</i>	119
Gambar 4.123 Tampilan status data barang <i>inactive</i>	120
Gambar 4.124 Tampilan PDF <i>Master Data Barang</i>	120
Gambar 4.125 Tampilan Halaman <i>Master Data Jenis Barang</i>	121
Gambar 4.126 Tampilan Halaman Mengubah Data Jenis Barang.....	121
Gambar 4.127 Tampilan Halaman Mnegubah Data Jenis Barang.....	122
Gambar 4.128 Tampilan notifikasi data sudah ada pada halaman <i>master data jenis barang</i>	122
Gambar 4.129 Tampilan notifikasi konfirmasi hapus data jenis barang.....	122
Gambar 4.130 Tampilan PDF <i>Master Data Jenis Barang</i>	123
Gambar 4.131 Tampilan Halaman <i>Master Data Satuan</i>	123
Gambar 4.132 Tampilan Halaman Mengubah Data Satuan.....	124
Gambar 4.133 Tampilan Halaman Mengubah Data Satuan.....	124
Gambar 4.134 Tampilan notifikasi data sudah ada pada halaman <i>master data satuan</i>	124
Gambar 4.135 Tampilan notifikasi konfirmasi hapus data satuan.....	125
Gambar 4.136 Tampilan PDF <i>Master Data Satuan</i>	125
Gambar 4.137 Tampilan Halaman <i>Master Data Pesanan</i>	126
Gambar 4.138 Tampilan PDF <i>Master Data Pesanan</i>	127
Gambar 4.139 Tampilan Halaman <i>Master Data Pesanan Detail</i>	127
Gambar 4.140 Tampilan Halaman <i>Master Data Meja</i>	127
Gambar 4.141 Tampilan Halaman Menambah Data Meja.....	128
Gambar 4.142 Tampilan Halaman Mengubah Data Meja.....	128
Gambar 4.143 Tampilan notifikasi data sudah ada pada halaman <i>master data meja</i>	129
Gambar 4.144 Tampilan notifikasi konfirmasi hapus data meja.....	129
Gambar 4.145 Tampilan PDF <i>Master Data Meja</i>	129
Gambar 4.146 Tampilan Halaman <i>Master Data Metode Bayar</i>	130
Gambar 4.147 Tampilan Halaman Menambah Data Metode Bayar.....	130
Gambar 4.148 Tampilan Halaman Mengubah Data Metode Bayar.....	131
Gambar 4.149 Tampilan notifikasi data sudah ada pada halaman <i>master data metode bayar</i>	131
Gambar 4.150 Tampilan notifikasi konfirmasi hapus data metode bayar.....	131
Gambar 4.151 Tampilan PDF <i>Master Data Metode Bayar</i>	132
Gambar 4.152 Tampilan Halaman <i>Master Data Pelanggan</i>	132
Gambar 4.153 Tampilan Halaman Menambah Data Pelanggan.....	133
Gambar 4.154 Tampilan Halaman Mengubah Data Pelanggan.....	133
Gambar 4.155 Tampilan notifikasi data sudah ada pada halaman <i>master data pelanggan</i>	133
Gambar 4.156 Tampilan notifikasi konfirmasi hapus data pelanggan.....	134
Gambar 4.157 Tampilan PDF <i>Master Data Pelanggan</i>	134
Gambar 4.158 Tampilan Halaman <i>Master Data Pengeluaran</i>	135
Gambar 4.159 Tampilan Halaman Menambah Data Pengeluaran.....	136

Gambar 4.160 Tampilan Halaman Mengubah Data Pengeluaran	136
Gambar 4.161 Tampilan notifikasi data sudah ada pada halaman <i>master data</i> pengeluaran	137
Gambar 4.162 Tampilan PDF <i>Master Data</i> Pengeluaran	137
Gambar 4.163 Tampilan Halaman <i>Master Data Supplier</i>	138
Gambar 4.164 Tampilan Halaman Menambah Data <i>Supplier</i>	138
Gambar 4.165 Tampilan Halaman Mengubah Data <i>Supplier</i>	139
Gambar 4.166 Tampilan notifikasi data sudah ada pada halaman <i>master data supplier</i>	139
Gambar 4.167 Tampilan notifikasi konfirmasi hapus data <i>supplier</i>	139
Gambar 4.168 Tampilan PDF <i>Master Data Supplier</i>	140
Gambar 4.169 Tampilan Halaman <i>Dashboard</i> Kasir.....	140
Gambar 4.170 Tampilan <i>Card</i> Pesanan Status Makanan Belum Dianter	141
Gambar 4.171 Tampilan <i>Card</i> Pesanan Status Beberapa Makanan Sudah Ada	141
Gambar 4.172 Tampilan <i>Card</i> Pesanan Status Semua Makanan Sudah Ada.....	141
Gambar 4.173 Tampilan <i>Card</i> Pesanan Status Selesai	141
Gambar 4.174 Tampilan Halaman Menambah Data <i>Pesanan</i>	142
Gambar 4.175 Tampilan Halaman Mengelola <i>Detail Pesanan</i>	143
Gambar 4.176 Tampilan Status <i>Detail Pesanan</i>	143
Gambar 4.177 Tampilan notifikasi pesanan lagi dimasak pada halaman <i>detail</i> pesanan	143
Gambar 4.178 Tampilan Halaman <i>Input Data Detail Pesanan</i>	144
Gambar 4.179 Tampilan notifikasi pesanan sudah selesai pada halaman <i>detail</i> pesanan	144
Gambar 4.180 Tampilan notifikasi bahan tidak cukup pada halaman <i>detail</i> pesanan.....	144
Gambar 4.181 Tampilan Halaman <i>Add Diskon</i>	144
Gambar 4.182 Tampilan <i>Bill</i> Penagihan	145
Gambar 4.183 Tampilan <i>Bill</i> Pelunasan.....	145
Gambar 4.184 Tampilan Halaman Pembayaran.....	146
Gambar 4.185 Tampilan notifikasi pembayaran tidak cukup.....	146
Gambar 4.186 Tampilan notifikasi pesanan sudah selesai pada halaman pembayaran.....	146
Gambar 4.187 Tampilan Halaman Pengecekan Data Pesanan.....	147
Gambar 4.188 Tampilan Halaman <i>Dashboard</i> Dapur	147
Gambar 4.189 Tampilan status <i>detail</i> pesanan pada halaman <i>dashboard</i> dapur.....	148
Gambar 4.190 Tampilan notifikasi pesanan belum diterima pada halaman <i>dashboard</i> dapur.....	148
Gambar 4.191 Tampilan notifikasi pesanan sudah diterima pada halaman <i>dashboard</i> dapur	148
Gambar D.1 Halaman <i>Log In</i>	1
Gambar D.2 Halaman <i>Sign Up</i>	2
Gambar D.3 Halaman <i>Dashboard</i> Manajer	2
Gambar D.4 Halaman <i>Side-Nav Master Data</i>	3
Gambar D.5 Halaman <i>Master Data</i>	3
Gambar D.6 Halaman Tambah Data	4
Gambar D.7 Notifikasi Duplikasi Data	4
Gambar D.8 Halaman PDF <i>Master Data</i>	4
Gambar D.9 Notifikasi Konfirmasi Deaktivasi Data	4
Gambar D.10 Halaman <i>Master Data User</i>	5
Gambar D.11 Halaman	5
Gambar D.12 Halaman PDF <i>Master Data User</i>	6
Gambar D.13 Halaman <i>Master Data Menu</i>	6
Gambar D.14 Halaman Tambah Data <i>Menu</i>	7
Gambar D.15 Halaman PDF <i>Master Data Menu</i>	7
Gambar D.16 Halaman <i>Master Data Menu Detail</i>	7
Gambar D.17 Halaman Tambah Data <i>Menu Detail</i>	8
Gambar D.18 Halaman <i>Master Data</i> Barang.....	8
Gambar D.19 Halaman Tambah Data Barang.....	9
Gambar D.20 Halaman PDF <i>Master Data</i> Barang.....	9
Gambar D.21 Halaman Tambah Jumlah Stok Inventori Barang	10
Gambar D.22 Halaman <i>Master Data Pesanan</i>	10
Gambar D.23 Halaman PDF <i>Master Data Pesanan</i>	11
Gambar D.24 Halaman <i>Master Data Pesanan Detail</i>	11
Gambar D.25 Halaman <i>Master Data Pengeluaran</i>	12

Gambar D.26 Halaman Tambah Pengeluaran	12
Gambar D.27 Halaman PDF <i>Master Data</i> Pengeluaran	13
Gambar D.28 Halaman <i>Dashboard</i> Kasir	13
Gambar D.29 Status Pesanan	14
Gambar D.30 Halaman Tambah Pesanan	15
Gambar D.31 Halaman Data <i>Menu</i> Pada Pesanan	15
Gambar D.32 Halaman Tambah <i>Menu</i> Pada Pesanan.....	16
Gambar D.33 Halaman Pemberian Diskon	16
Gambar D.34 Halaman <i>Dashboard</i> Dapur.....	17
Gambar D.35 Halaman <i>Bill</i> Penagihan	17
Gambar D.36 Halaman Pembayaran.....	18
Gambar D.37 Halaman <i>Bill</i> Pelunasan.....	18
Gambar E.1 Observasi Pada Restoran Ta Chia	1
Gambar E.2 Penampilan Sistem Pada Restoran Ta Chia	2
Gambar E.3 Implementasi Sistem Pada Restoran Ta Chia	3



DAFTAR TABEL

Tabel 2.1 Elemen-elemen pada <i>class diagram</i>	17
Tabel 2.2 Elemen-elemen pada <i>use case diagram</i>	18
Tabel 2.3 Elemen-elemen pada <i>activity diagram</i>	20
Tabel 2.4 Penelitian terdahulu.....	22
Tabel 4.1 Spesifikasi Perangkat Milik Pembangun Sistem Usulan.....	36
Tabel 4.2 Spesifikasi Perangkat Milik Restoran Ta Chia.....	37
Tabel 4.3 Deskripsi Tabel <i>User</i>	69
Tabel 4.4 Deskripsi Tabel <i>Role</i>	69
Tabel 4.5 Deskripsi Tabel <i>Barang</i>	70
Tabel 4.6 Deskripsi Tabel <i>Jenis Barang</i>	70
Tabel 4.7 Deskripsi Tabel <i>Satuan</i>	71
Tabel 4.8 Deskripsi Tabel <i>Menu</i>	71
Tabel 4.9 Deskripsi Tabel <i>Menu Detail</i>	71
Tabel 4.10 Deskripsi Tabel <i>Kategori</i>	72
Tabel 4.11 Deskripsi Tabel <i>Pengeluaran</i>	72
Tabel 4.12 Deskripsi Tabel <i>Supplier</i>	72
Tabel 4.13 Deskripsi Tabel <i>Pesanan</i>	73
Tabel 4.14 Deskripsi Tabel <i>Pesanan Detail</i>	73
Tabel 4.15 Deskripsi Tabel <i>Meja</i>	74
Tabel 4.16 Deskripsi Tabel <i>Metode Bayar</i>	74
Tabel 4.17 Deskripsi Tabel <i>Pelanggan</i>	75
Tabel 4.18 <i>Script #1.1 Log In</i>	151
Tabel 4.19 <i>Script #1.2 Log Out</i>	151
Tabel B.1 <i>Use Case Description Log In</i>	1
Tabel B.2 <i>Use Case Description Sign Up</i>	2
Tabel B.3 <i>Use Case Description Mengelola User</i>	3
Tabel B.4 <i>Use Case Description Mengelola Role</i>	4
Tabel B.5 <i>Use Case Description Mengelola Menu</i>	5
Tabel B.6 <i>Use Case Description Mengelola Detail Menu</i>	6
Tabel B.7 <i>Use Case Description Mengelola Kategori Menu</i>	7
Tabel B.8 <i>Use Case Description Inventori Barang</i>	8
Tabel B.9 <i>Use Case Description Mengelola Stok Inventori Barang</i>	9
Tabel B.10 <i>Use Case Description Mengelola Jenis Barang</i>	10
Tabel B.11 <i>Use Case Description Mengelola Satuan Barang</i>	11
Tabel B.12 <i>Use Case Description Mengelola Pemasukan</i>	12
Tabel B.13 <i>Use Case Description Mengelola Meja</i>	13
Tabel B.14 <i>Use Case Description Mengelola Metode Pembayaran</i>	14
Tabel B.15 <i>Use Case Description Mengelola Pelanggan</i>	15
Tabel B.16 <i>Use Case Description Mengelola Pengeluaran</i>	16
Tabel B.17 <i>Use Case Description Mengelola Supplier</i>	17
Tabel B.18 <i>Use Case Description Mengelola Pesanan Pelanggan</i>	18
Tabel B.19 <i>Use Case Description Mengelola Detail Pesanan Pelanggan</i>	19
Tabel B.20 <i>Use Case Description Melakukan Konfirmasi Pesanan</i>	20
Tabel B.21 <i>Use Case Description Memberikan Diskon Pesanan</i>	21
Tabel B.22 <i>Use Case Description Bayar Pesanan</i>	22
Tabel C.1 <i>Script #2.1 Sign Up</i>	2
Tabel C.2 <i>Script #3.1 Menambah Data User Baru</i>	4
Tabel C.3 <i>Script #3.2 Mengubah Data User</i>	5
Tabel C.4 <i>Script #3.3 Mendeaktivasikan User</i>	6
Tabel C.5 <i>Script #3.4 Memilih PDF User</i>	6
Tabel C.6 <i>Script #4.1 Menambah Data Role Baru</i>	8
Tabel C.7 <i>Script #4.2 Mengubah Data Role</i>	9
Tabel C.8 <i>Script #4.3 Mendeaktivasikan Role</i>	10
Tabel C.9 <i>Script #4.4 Memilih PDF Role</i>	10
Tabel C.10 <i>Script #5.1 Menambah Data Menu Baru</i>	12

Tabel C.11 <i>Script #5.2 Mengubaha Data Menu</i>	13
Tabel C.12 <i>Script #5.3 Mendeaktivasikan Menu</i>	14
Tabel C.13 <i>Script #5.4 Mengaktivasikan Menu</i>	14
Tabel C.14 <i>Script #5.5 Memilih PDF Menu</i>	14
Tabel C.15 <i>Script #6.1 Memilih Data Detail Menu</i>	16
Tabel C.16 <i>Script #6.2 Menambah Data Detail Menu</i>	16
Tabel C.17 <i>Script #6.3 Mengubah Data Detail Menu</i>	17
Tabel C.18 <i>Script #6.4 Menghapus Detail Menu</i>	18
Tabel C.19 <i>Script #7.1 Menambah Data Kategori Baru</i>	20
Tabel C.20 <i>Script #7.2 Mengubah Data Kategori</i>	21
Tabel C.21 <i>Script #7.3 Mendeaktivasikan Kategori</i>	22
Tabel C.22 <i>Script #7.4 Memilih PDF Kategori</i>	22
Tabel C.23 <i>Script #8.1 Menambah Data Inventori Baru</i>	24
Tabel C.24 <i>Script #8.2 Mengubah Data Inventori</i>	25
Tabel C.25 <i>Script #8.3 Menambah Jumlah Stok Barang</i>	26
Tabel C.26 <i>Script #8.4 Mendeaktivasikan Inventori</i>	26
Tabel C.27 <i>Script #8.5 Mengaktivasikan Inventori</i>	26
Tabel C.28 <i>Script #8.6 Memilih PDF Inventori</i>	27
Tabel C.29 <i>Script #9.1 Menambah Data Jenis Barang Baru</i>	29
Tabel C.30 <i>Script #9.2 Mengubah Data Jenis Barang</i>	30
Tabel C.31 <i>Script #9.3 Mendeaktivasikan Jenis Barang</i>	31
Tabel C.32 <i>Script #9.4 Memilih PDF Jenis Barang</i>	31
Tabel C.33 <i>Script #10.1 Menambah Data Satuan Baru</i>	33
Tabel C.34 <i>Script #10.2 Mengubah Data Satuan</i>	34
Tabel C.35 <i>Script #10.3 Mendeaktivasikan Satuan</i>	34
Tabel C.36 <i>Script #10.4 Memilih PDF Satuan</i>	35
Tabel C.37 <i>Script #11.1 Sorting Data Pemasukan</i>	36
Tabel C.38 <i>Script #11.2 Memilih Data Detail Pesanan</i>	37
Tabel C.39 <i>Script #11.3 Memilih PDF Pemasukan</i>	37
Tabel C.40 <i>Script #12.1 Menambah Data Meja Baru</i>	39
Tabel C.41 <i>Script #12.2 Mengubah Data Meja</i>	40
Tabel C.42 <i>Script #12.3 Mendeaktivasikan Meja</i>	41
Tabel C.43 <i>Script #12.4 Memilih PDF Meja</i>	41
Tabel C.44 <i>Script #13.1 Menambah Data Metode Bayar Baru</i>	43
Tabel C.45 <i>Script #13.2 Mengubah Data Metode Bayar</i>	44
Tabel C.46 <i>Script #13.3 Mendeaktivasikan Metode Bayar</i>	45
Tabel C.47 <i>Script #13.4 Memilih PDF Metode Bayar</i>	45
Tabel C.48 <i>Script #14.1 Menambah Data Pelanggan Baru</i>	47
Tabel C.49 <i>Script #14.2 Mengubah Data Pelanggan</i>	48
Tabel C.50 <i>Script #14.3 Mendeaktivasikan Pelanggan</i>	48
Tabel C.51 <i>Script #14.4 Memilih PDF Pelanggan</i>	49
Tabel C.52 <i>Script #15.1 Sorting Data Pengeluaran</i>	50
Tabel C.53 <i>Script #15.2 Menambah Data Pengeluaran Baru</i>	51
Tabel C.54 <i>Script #15.3 Mengubah Data Pengeluaran</i>	51
Tabel C.55 <i>Script #15.4 Memilih PDF Pengeluaran</i>	52
Tabel C.56 <i>Script #16.1 Menambah Data Supplier</i>	54
Tabel C.57 <i>Script #16.2 Mengubah Data Supplier</i>	55
Tabel C.58 <i>Script #16.3 Mendeaktivasikan Supplier</i>	56
Tabel C.59 <i>Script #16.4 Memilih PDF Supplier</i>	56
Tabel C.60 <i>Script #17.1 Mengelola Pesanan Pelanggan</i>	58
Tabel C.61 <i>Script #18.1 Menambah Menu Ke Dalam Pesanan</i>	60
Tabel C.62 <i>Script #18.2 Memberikan Diskon</i>	61
Tabel C.63 <i>Script #18.3 Membatalkan Menu Pada Pesanan</i>	61
Tabel C.64 <i>Script #18.4 Menyajikan Menu Pada Pesanan</i>	62
Tabel C.65 <i>Script #19.1 Menerima Pesanan Baru</i>	64
Tabel C.66 <i>Script #19.2 Menyelesaikan Pesanan</i>	64
Tabel C.67 <i>Script #20.1 Memilih Bill Penagihan</i>	66

Tabel C.68 *Script #20.2* Melakukan Pembayaran..... 66
Tabel C.69 *Script #20.3* Memilih *Bill* Pelunasan..... 67



BAB I

PENDAHULUAN

1.1. Latar Belakang

Pada zaman sekarang banyak usaha bisnis sudah menggunakan teknologi atau mesin untuk membantu pekerjaan sehari-hari mereka. Salah satunya adalah usaha bisnis kuliner. Penggunaan teknologi ini dilakukan untuk mendapatkan berbagai keuntungan seperti mempertahankan kualitas pelayanan, meningkatkan produksi kinerja, mempermudah pemasaran, mengurangi biaya pengelolaan, dan meningkatkan keuntungan usaha[1]. Masih ada banyak lagi keuntungan yang bisa didapatkan, tetapi hal yang paling penting adalah cara memperoleh data yang diinginkan dan dikelola menjadi sebuah informasi.

Saat ini sudah banyak restoran dan *café* yang sudah menggunakan dan memanfaatkan sistem informasi dalam proses bisnis mereka. Maupun itu dengan menggunakan sebuah komputer berbasis *web* atau merancang sebuah aplikasi online. Dengan adanya sistem informasi, semua pekerjaan dan manajemen data yang diperlukan akan terkelola dengan efisien dan efektif. Begitu juga dalam memperoleh dan mengelola data-data usaha [2]. Itulah mengapa banyak perusahaan sekarang sudah melakukan komputersasi pada bisnisnya. Dibandingkan dengan pengolahan data secara manual seperti menggunakan kertas, menggunakan komputer atau alat elektronik lainnya akan lebih cepat untuk dilakukan.

Restoran Ta Chia adalah salah satu usaha kuliner yang menyediakan masakan *Chinese food* yang non-halal. Proses pencatatan dan pengelolaan data masih dilakukan secara manual via catat kertas. Begitu juga dengan pemberian pesanan yang diberikan dari kasir kepada pihak dapur atau bartender yang masih memerlukan seorang pelayan sebagai perantara kedua pihak. Pelaksanaan kegiatan tersebut menghambat pekerjaan menjadi tidak maksimal. Selain tidak maksimalnya kinerja usaha, akan ada juga kesalahan yang bisa terjadi seperti hilang atau terhapusnya data, salah penulisan data, serta perhitungan keuangan yang tidak akurat. Masalah-masalah ini dapat diatasi dengan merancang sebuah sistem informasi yang dapat membantu mencatat dan mengolah data sehingga dapat membantu meringankan beban pengumpulan data. Tujuan yang ingin dicapai dari tugas akhir ini adalah merancang sebuah sistem pencatatan data keuangan dan inventori berbasis *web* untuk Restoran Ta Chia.

1.2. Rumusan Masalah

Rumusan masalah yang terdapat pada penelitian adalah sebagai berikut:

- 1) Bagaimana cara merancang aplikasi berbasis *website* untuk restoran?
- 2) Bagaimana sistem yang dirancang dapat mempermudah dan meringankan beban restoran?

1.3. Batasan Masalah

Batasan masalah yang terdapat pada penelitian adalah sebagai berikut:

- 1) Sistem hanya dapat digunakan untuk melakukan proses pengelolaan keuangan dan inventori usaha. Keuangan meliputi pemasukan dan pengeluaran. Inventori meliputi bahan baku dan peralatan.
- 2) Sistem tidak dapat digunakan oleh pihak konsumen, yakni pembelian makanan.

1.4. Tujuan dan Manfaat Penelitian

Berikut merupakan tujuan dan manfaat dari penelitian ini:

1.4.1. Tujuan Penelitian

Tujuan dari penelitian ini adalah untuk mengembangkan sebuah sistem aplikasi berbasis web untuk restoran usaha kecil dan mengimplementasikannya pada proses bisnis yang difokuskan pada pengelolaan data keuangan dan inventoris.

1.4.2. Manfaat Penelitian

Manfaat dari penelitian ini adalah untuk mempermudah proses bisnis usaha restoran dalam pencatatan keuangan dan inventoris usaha agar dapat meningkatkan efisiensi dan efektivitas berjalannya usaha.

1.5. Metodologi Penelitian

Berikut merupakan metodologi penelitian yang digunakan dari penelitian:

1.5.1. Metodologi Pengumpulan Data

Metode pada pengumpulan data menggunakan metode wawancara, metode studi literatur, dan metode survei. Metode wawancara dilakukan untuk mengetahui proses apa saja yang sekarang lagi dijalankan pada kegiatan kerja mereka sehari-hari. Wawancara akan dilakukan dengan pemilik usaha restoran dan beberapa karyawan yang bekerja disana. Metode studi literatur dilakukan untuk mencari tahu apa saja teori yang perlu digunakan dalam merancang sistem administrasi pada perusahaan restoran. Studi literatur akan dilakukan melalui jurnal-jurnal dan artikel-artikel penelitian yang sudah ada.

1.5.2. Metode Pengembangan Sistem

Metode pada pengembangan sistem informasi menggunakan metode prototyping. Metode ini diawali dengan tahap *planning* yang kemudian dilanjutkan dengan tahap selanjutnya yang memiliki 3 fase yang dilakukan secara bersamaan yaitu *analysis*, *design*, dan *implementation*. Ketiga tahap tersebut menghasilkan sebuah sistem prototype yang kemudian ditunjukkan kepada pengguna yang akan memberikan komentar dan masukan. Proses metode ini akan diulang kepada 4 fase sebelumnya dan dilakukan berulang-ulang sampai sistem disetujui oleh pengguna. Metode ini memberikan keuntungan yaitu sangat cepat menyediakan sistem dimana pengguna dapat berinteraksi dengan sistem tersebut.

1.6. Sistematika Penulisan

Berikut merupakan sistematika penulisan dari penelitian ini:

BAB 1: PENDAHULUAN

Bab ini mengandung latar belakang, rumusan masalah, batasan masalah, tujuan dan manfaat penelitian, metodologi penelitian, dan sistematika penulisan.

BAB 2: LANDASAN TEORI

Bab ini mengandung teori-teori dasar yang dikutip dari beberapa buku, jurnal, dan artikel penelitian yang berhubungan dengan penelitian ini.

BAB 3: SISTEM SAAT INI

Bab ini mengandung pembahasan mengenai sistem yang digunakan pada saat ini, serta kendala yang terdapat pada sistem tersebut.

BAB 4: ANALISIS DAN PERANCANGAN SISTEM USULAN

Bab ini mengandung pembahasan perencanaan yang mengandung identifikasi dan analisis kelayakan sistem, serta tahap analisis *user requirement* berupa *use case diagram*, *activity diagram*, dan *class diagram*. Bab ini juga mengandung tahap perancangan *database* berupa *table relationship diagram* dan *table description*, serta perancangan *interface* sistem.

BAB 5: KESIMPULAN DAN SARAN

Bab ini mengandung kesimpulan dan saran mengenai sistem yang akan dirancang dari penelitian ini.

BAB II

LANDASAN TEORI

2.1. Kajian Teori

Berikut merupakan kajian teori yang didasarkan dari artikel, literatur, dan sumber yang terkait dengan tema penelitian ini

2.1.1. Restoran

Restoran dipandang sebagai tempat dimana orang kunjungi untuk melepaskan badan dari rasa lelah dan lapar. Pernyataan tersebut telah membagi perspektif orang menjadi dua mengenai apa arti sesungguhnya dari restoran. Perspektif pertama mengatakan bahwa restoran adalah sebuah tempat yang menyediakan menu-menu makanan dan minuman untuk memenuhi atau memuaskan kebutuhan para pelanggan dari rasa kelaparan. Perspektif kedua mengatakan bahwa restoran adalah sebuah tempat dimana para pelanggan kunjungi untuk melepaskan kelelahan yang dibebani dengan aktivitas-aktivitas luar. Dinyatakan juga mengenai pengertian dari restoran, yaitu dipandang sebagai tempat dimana pelanggan ingin melakukan atau mengalami interaksi secara sosial dan juga secara kultural. Pernyataan tersebut menekankan bahwa restoran adalah sebuah usaha yang bukan saja menyediakan sajian makanan dan minuman untuk memenuhi kepuasan pengunjung, tetapi juga menyediakan suasana yang nyaman dan aman agar para pengunjung dapat melakukan interaksi sosial dan kultural kepada sesama [3].

Restoran diklasifikasikan dengan berbagai cara. Faktor utama yang membedakan sebuah restoran adalah jenis makanan yang disajikan dan gaya layanan yang diberikan. Berdasarkan jenis layanan yang diberikan, restoran dibagi menjadi 3 tipe restoran, sebagai berikut [4].

- *Fast food restaurant*

Restoran yang telah menyediakan makanan cepat saji kepada pelanggan dengan menu dan harga yang terbatas. Makanan sudah dimasak terlebih dahulu dengan jumlah yang banyak dan disimpan dalam tempat dengan suhu yang sesuai. Restoran cepat saji memungkinkan para pelanggannya untuk langsung mengambil makanannya sendiri lewat kasir atau drive thru dengan cepat.

- *Casual dining*

Restoran yang menyediakan menu makanan yang cukup banyak dan memberikan pelayanan kepada hampir semua jenis pelanggan. Harga makanan yang ditetapkan pada biasanya tidak terlalu tinggi sehingga dapat dinikmati oleh semua pelanggan.

- *Fine dining*

Restoran yang menyediakan menu makanan yang lebih spesifik dan khusus kepada para pelanggannya, tergantung jenis atau tema makanan yang disajikan. Suasana restoran terlihat berkelas dengan dekorasi-dekorasi yang berkualitas tinggi. Terdapat *waiter* atau *waitress* yang melayani pelanggan dengan terlatih dan berpakaian formal. Terdapat juga *chef* yang memberikan sentuhan profesional kepada masakan yang disajikan. Harga yang ditetapkan pada biasanya cukup tinggi dan tiap menu terkadang memiliki aturan tertentu yang perlu diikuti untuk menikmatinya.

2.1.2. **Komputerisasi**

Komputerisasi adalah proses perubahan proses pengolahan data, yang dimana sebelumnya dilakukan secara manual digantikan dengan bantuan media komputer sehingga dapat dilakukan dengan otomatis. Cara manual yang dimaksud salah satunya adalah pencatatan data menggunakan kertas. Berbeda dengan komputer, komputerisasi merupakan perubahan proses, sedangkan komputer adalah alat yang digunakan dalam proses komputerisasi. Komputer adalah sebuah alat atau perangkat elektronik yang pada biasanya digunakan pada aktivitas sehari-hari untuk memudahkan pekerjaan yang dilakukan sehingga menjadi lebih cepat dan akurat.

Komputerisasi dapat sangat bermanfaat di berbagai bidang seperti bidang pemerintahan, kesehatan, industri pangan, perekonomian, dan lain-lain. Dengan adanya komputerisasi, perusahaan dapat meningkatkan pengelolaan data dan informasi mereka dengan lebih baik sehingga dapat membantu dalam proses pengampilan keputusan untuk mengembangkan sistem yang sudah ada atau sistem yang baru [5].

2.1.3. **Inventory**

Inventory atau persediaan secara harafiah adalah persediaan bahan baku atau barang jadi yang perlu disimpan untuk keperluan usaha. Secara umum, *inventory* atau persediaan barang adalah aset-aset usaha yang belum diproses yang disimpan dan kemudian dijual kepada pelanggan sesuai dengan permintaannya untuk mendapatkan keuntungan [6].

2.1.4. Sistem Informasi Keuangan

Sistem informasi keuangan adalah sebuah sistem informasi yang dibuat untuk menunjukkan aliran uang yang terdapat pada perusahaan. Informasi tersebut dapat digunakan untuk membantu menyelesaikan masalah keuangan usaha. Pada biasanya sebuah perusahaan memiliki seorang auditor yang bertanggung jawab untuk merawat keutuhan sistem keuangan perusahaan [7].

Pada setiap pencatatan keuangan, hasil akhir yang terbuat adalah laporan keuangan untuk mencatat masuk keluarnya arus kas perusahaan. Laporan keuangan dapat dibuat secara manual ataupun digital. Membuat dengan manual dilakukan dengan cara memakai tulis tangan pada kertas atau buku catatan besar. Membuat dengan digital dilakukan dengan menyimpan data pada komputer seperti Excel [8].

2.1.5. Web Programming

Pemrograman *web* mengacu pada penulisan, markup, dan pengodingan yang melibatkan pengembangan *web*, yang mencakup klien, skrip, dan keamanan jaringan. Bahasa yang paling umum digunakan pada pemrograman *web* adalah XML, HTML, Perl 5, JavaScript, dan PHP. Pada penelitian ini, pengembangan *website* akan menggunakan bahasa pemrograman PHP dan JavaScript, dan menggunakan markup HTML dan CSS.

2.1.5.1. PHP

PHP merupakan bahasa pemrograman yang terpasang pada HTML dan digunakan untuk mengembangkan suatu *website*. Singkatan PHP pada mulanya adalah *Personal Home Page* (situs personal), tetapi dengan seiring berjalannya waktu singkatan PHP diubah menjadi *Hypertext Preprocessor*. *Syntax* yang digunakan kebanyakan dari bahasa pemrograman lain seperti C, Java, dan Perl. Pada biasanya PHP digunakan untuk mengekstrak data yang ada di *database* suatu server web dan ditampilkan pada sebuah *webpage* [7].

Kelebihan yang didapatkan dari menggunakan PHP dalam mengembangkan *website* adalah berikut [9]:

- PHP bersifat *open source*, sehingga dapat diakses oleh semua orang yang memiliki keinginan untuk melakukan pemrograman *website* tanpa memiliki lisensi khusus untuk membuatnya.
- *Script* yang dimiliki PHP dapat dijalankan pada server *web* dan berbagai jenis operasi sistem seperti LINUX, Windows, dan Mac.
- *Script* PHP dapat dimasukkan ke dalam *tag* HTML.
- Kecepatan akses penggunaan PHP cukup tinggi.

2.1.5.2. Codeigniter

CodeIgniter adalah sebuah *framework* PHP yang bersifat *open source*. Biasanya digunakan pada pemrograman PHP untuk membantu merancang sebuah *website* menjadi lebih cepat dan mudah. *CodeIgniter* menggunakan kerangka kerja MVC (*Model – View – Controller*), dimana kerangka tersebut sangat berguna untuk memisahkan data, presentasi, dan kerja logika [10]. Keunggulan dalam memakai *framework CodeIgniter* yaitu mudah dipelajari dan dikuasai, memiliki *library* yang dapat dimodifikasi, sangat responsiv, memiliki ukuran *byte* yang kecil, dan memiliki banyak fitur yang fleksibel sehingga tidak bergantung pada satu sama lain [11].

Kelebihan yang didapatkan dari menggunakan *CodeIgniter* dalam mengembangkan *website* adalah berikut [12]:

- *File* yang dimiliki berukuran kecil.
- Dokumentasi yang sudah cukup bagus.
- Kompatibel dengan berbagai macam *platform hosting website*.
- Kinerja yang baik.
- Mengintegrasikan penggunaan *Library*.

Kekurangan yang didapatkan dari menggunakan *CodeIgniter* dalam mengembangkan *website* adalah berikut [12]:

- Tidak disarankan untuk digunakan untuk membuat proyek berskala besar.
- Meskipun dapat menggunakan *Library*, penggunaannya masih terbatas karena sulit untuk mencari *plugin*.
- Belum ada *editor* yang khusus untuk digunakan dengan *CodeIgniter*.

2.1.5.3. HTML

HTML merupakan singkatan dari *HyperText Markup Language*, sebuah bahasa yang digunakan secara umum pada perancangan sebuah dokumen *webpage* atau *website*. HTML bukanlah sebuah bahasa pemrograman tetapi bahasa markup, yang berarti adalah sebuah sistem yang digunakan untuk mengidentifikasi dan mendeskripsikan berbagai macam komponen pada dokumen seperti heading, paragraf, dan list. Tiap *file* HTML memiliki ekstensi berupa *.html* atau *.htm* [13].

Struktur kode yang terdapat pada HTML terdiri dari 4 struktur utama:

- `<!DOCTYPE HTML>`
Tag yang mengawali dokumen HTML, berfungsi sebagai pemberi informasi kepada server bahwa *file* atau dokumen yang dibuka adalah *file* HTML.
- `<html>...</html>`
Tag yang mengawali isi dari *file* HTML
- `<head>...</head>`
Tag yang dijadikan sebagai kepala dan berfungsi sebagai judul pada *webpage*.
- `<body>...</body>`
Tag yang dijadikan sebagai badan dan berfungsi sebagai isi pada *webpage*. Isi yang dimaksud adalah tulisan, gambar, dan isi apapun yang ditampilkan pada suatu *webpage*.

2.1.5.4. CSS

CSS merupakan singkatan dari *Cascading Style Sheets*, sebuah bahasa *style sheets* yang digunakan untuk memasukan atau memodifikasikan tampilan yang dipresentasikan pada suatu *website*. Tampilan yang dapat diubah dapat berupa *font size*, *font style*, *color*, *background*, *layout*, dan lain-lain. Beda dengan HTML yang hanya bisa menampilkan isi dari sebuah *website*, bisa dikatakan CSS memberikan warna untuk memperindah tampilan isi *website* menjadi lebih menarik. *File* CSS merupakan *tool* yang sangat berguna karena bisa menggunakan hanya satu *file* CSS untuk mengubah tampilan semua *webpage* [13].

2.1.5.5. Javascript

JavaScript merupakan bahasa *scripting* yang digunakan untuk memberikan interaktivitas dan perilaku pada sebuah *website*. Perilaku yang dimaksud berupa

memeriksa validitas pengisian form, pembuatan *interface* menu, membuat sebuah browser mengingat user yang telah menggunakannya, dan lain-lain. Biasanya digunakan untuk memanipulasi elemen-elemen yang ada pada *webpage*, *style*, ataupun browser-nya sendiri. Dibandingkan dengan HTML dan CSS, menggunakan JavaScript memiliki sedikit persamaan dengan programming maka diperlukan pengalaman dalam menggunakan bahasa pemrograman seperti Java atau C [13].

2.1.6. Sistem Basis Data (*Database*)

Database atau basis data adalah kumpulan data-data yang telah dikumpulkan dan disusun dalam bentuk tabel, dimana tabel-tabel tersebut dapat berdiri sendiri atau saling berhubungan dengan satu sama lain pada suatu media [14]. *Database* pada umumnya digunakan untuk menyimpan data, tetapi bukan hanya itu saja yang dapat dilakukan pada basis data. Selain menyimpan data, pengguna basis data juga dapat menghapus data, menggantikan data, dan menggunakan fungsi-fungsi lainnya yang dapat digunakan untuk membantu pengguna dalam mengelola data dalam *database*. Fungsi-fungsi tersebut adalah sebagai berikut [15]:

- *Data dictionary management*: Sistem basis data yang menyimpan elemen-elemen data dan hubungannya dengan data lain dalam sebuah kamus data.
- *Data storage management*: Sistem basis data membuat dan mengelola struktur kompleks yang diperlukan dalam penyimpanan data, sehingga pengguna tidak perlu mendefinisikan dan memprogram karakteristik data fisik.
- *Data transformation and presentation*: Sistem basis data mengubah data yang dimasukkan agar sesuai dengan struktur data yang diperlukan pengguna. Sistem basis data memformatkan data yang diambil secara fisik untuk membuat data tersebut menjadi sesuai dengan apa yang diharapkan pengguna.
- *Security management*: Sistem basis data membuat sebuah sistem keamanan yang menegakkan keamanan dan privasi data pengguna. Aturan yang dibuat menentukan pengguna mana yang dapat mengakses *database*, data yang mana saja pengguna dapat akses, dan operasi data apa saja yang pengguna dapat gunakan.
- *Multiuser access control*: Sistem basis data menyediakan integritas dan konsistensi data dengan cara menggunakan sebuah algoritma yang memastikan jika terdapat banyak pengguna mengakses *database* secara bersamaan maka tidak mengurangi integritas.

- *Backup and recovery management*: Sistem basis data menyediakan percadangan dan pemulihan data untuk memastikan keamanan dan integritas data.
- *Data integrity management*: Sistem basis data menegakkan aturan integritas untuk meminimalisasikan redundansi data dan memaksimalkan konsistensi data.
- *Database access language and application programming interfaces*: Sistem basis data menyediakan cara mengakses data dengan menggunakan bahasa query.
- *Database communication interfaces*: Sistem basis data pada zaman sekarang menyediakan akses dari berbagai jaringan yang berbeda dengan internet melalui penggunaan browser, seperti Google Chrome, Mozilla Firefox, dan Microsoft Internet Explorer.

Berikut merupakan bagian-bagian dari *database* [15]:

- *Data*: Fakta mentah, seperti nomor telepon, nama orang, tempat tinggal, tanggal lahir, dan sebagainya. Data pada awalnya tidak memiliki arti apapun sebelum disusun menjadi sebuah informasi.
- *Table*: Kumpulan dari beberapa *field* dan *record*. Field ditunjukkan dalam bentuk kolom (*column*) pada table, sedangkan record ditunjukkan dalam bentuk baris (*row*).
- *Field*: Sebuah karakter atau kumpulan karakter, maupun itu huruf atau angka, yang memiliki arti yang tertentu. Satu *field* pada umumnya mewakili satu data mengenai banyak orang atau benda, salah satunya nama-nama pegawai suatu perusahaan.
- *Record*: Kumpulan dari beberapa data pada *field* yang saling berkaitan, sehingga membentuk suatu informasi mengenai suatu hal atau orang secara lengkap. Satu *record* pada umumnya mewakili satu data atau informasi mengenai seseorang, misalkan nama, tempat tinggal, email, dan lain-lain.

2.1.7. *Entity Relationship Diagram (ERD)*

Entity Relationship Diagram adalah model yang menunjukkan informasi yang dibuat, disimpan, dan digunakan oleh sistem bisnis. ERD dapat digunakan untuk menemukan informasi tiap komponen dalam suatu sistem dan bagaimana mereka diatur dan terkait dengan satu sama lain. Pada ERD, jenis informasi yang serupa didaftarkan secara bersama dan ditempatkan di dalam kotak yang disebut entitas. Sebuah entitas dihubungkan dengan entitas lain dengan sebuah garis yang menghubungkani mereka, dan terdapat sebuah simbol khusus yang ditambahkan ke diagram untuk

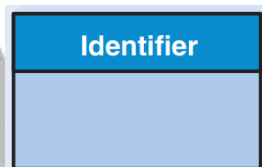
mengkomunikasikan aturan bisnis tingkat tinggi yang perlu didukung oleh sistem. ERD tidak memiliki urutan tetapi entitas yang terkait satu sama lain biasanya ditempatkan dengan dekat [16].

Berikut merupakan elemen-elemen yang digunakan dalam pembuatan ERD [16]:

- *Entity*

Entitas adalah model data yang sederhana berupa blok bangunan yang diartikan sebagai orang, tempat, atau peristiwa. Entitas digambarkan dalam bentuk persegi panjang, dan dijelaskan oleh kata benda tunggal dengan huruf kapital. Semua entitas memiliki nama, deskripsi singkat yang menjelaskan apa itu entitas, dan pengidentifikasi yang merupakan cara untuk menemukan informasi dalam entitas.

Berikut merupakan simbol dari *entity*:



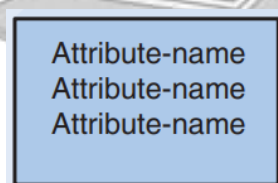
Gambar 2.1. Entity ERD

Sumber: Dennis, Alan, et al., System Analysis & Design, 5th ed. United States of America : Wiley, 2012.

- *Attribute*

Attribute adalah sebuah kumpulan dari satu atau berbagai karakteristik atau properti yang dipegang oleh sebuah *entity*. Penamaan *attribute* mendeskripsikan properti dan tipe yang mendeskripsikan jenis atributnya, seperti contoh *varchar* untuk *string*, sedangkan *int* untuk *integer*.

Berikut merupakan simbol dari *attribute*:



Gambar 2.2 Attribute ERD

Sumber: Dennis, Alan, et al., System Analysis & Design, 5th ed. United States of America : Wiley, 2012.

- *Relationship*

Relationship adalah hubungan yang terdapat pada dua atau berbagai entitas, menandakan bahwa entitas-entitas tersebut saling terkait satu sama lain. Setiap

relationship memiliki *parent entity* sebagai entitas pertama dan *child entity* sebagai entitas kedua. *Relationship* memiliki 2 properti yaitu kardinalitas dan modalitas.

Kardinalitas adalah sebuah rasio dari *parent entity* dengan *child entity*. Rasio tersebut dibagi menjadi 3 yaitu *one to one* (1:1), *one to many* (1:M), *many to many* (M:M).

Modalitas adalah hubungan yang menentukan *null* atau tidak *null*-nya sebuah hubungan yang mengacu pada apakah turunan dari *child entity* dapat ada atau tidak tanpa hubungan terkait di *parent entity*. Pada dasarnya, modalitas suatu hubungan menunjukkan apakah *child entity* diperlukan untuk berpartisipasi dalam sebuah hubungan.

2.1.8. System Development Life Cycle (SDLC)

SDLC merupakan singkatan dari *System Development Life Cycle*, sebuah proses yang digunakan untuk menentukan kegunaan atau manfaat suatu sistem informasi dalam mendukung proses kerjanya bisnis. Selain mendukung bisnis dapat juga digunakan untuk menentukan bagaimana suatu sistem dirancang, dibangun, dan dikirimkan kepada pengguna. Yang perlu diingatkan dari merancang sebuah sistem adalah sistem tersebut bukan hanya dibuat seindah atau sekagum mungkin tetapi dibuat agar dapat mendukung sebuah bisnis dan memberikan sebuah *value* kepada suatu organisasi. Dalam membangun suatu sistem informasi dengan menggunakan SDLC mengikuti 4 rangkaian fase utama yaitu tahap perencanaan, analisis, desain, dan implementasi [16].

2.1.8.1. Tahapan Perencanaan (Planning)

Fase perencanaan adalah tahap dimana sistem analis memahami tujuan atau alasan suatu sistem perlu dibangun. Fase ini memiliki 2 tahap utama yaitu tahap inisiasi proyek dan pengelolaan proyek. Pada tahap inisiasi proyek, sistem analis mengidentifikasi nilai bisnis pada perusahaan seperti cara menurunkan biaya dan meningkatkan pendapatan. Selain itu juga melakukan analisis kelayakan yang dibagi menjadi 3 bagian yaitu kelayakan teknis, kelayakan ekonomi, kelayakan organisasi. Pada tahap pengelolaan proyek, proyek sudah disetujui dan manajer proyek menentukan rencana kerja, tenaga kerja, dan memberikan arahan untuk menjalani proyek. Hasil yang didapatkan dari fase ini adalah *project plan* yang menjelaskan bagaimana tim proyek akan mengembangkan sistem informasi tersebut [16].

2.1.8.2. Tahapan Analisis (*Analysis*)

Fase analisis adalah tahap untuk Analisa pada sistem yang dibangun. Siapakah yang akan menggunakan sistem, apa saja yang bisa dilakukan oleh sistem, dimana dan kapan sistem akan digunakan. Pada fase ini, tim proyek menyelidiki sistem yang sudah ada dan mengidentifikasi peluang untuk mengembangkan sistem. Fase analisis memiliki 3 tahap [16]:

- Pembuatan strategi analisis untuk membangun panduan kepada tim proyek.
- Pengumpulan *requirement* seperti penyebaran kuesioner, wawancara, atau lokakarya kelompok.
- Pembuatan proposal yang akan ditunjukkan kepada proyek sponsor dan pemegang keputusan.

2.1.8.3. Tahapan Desain (*Design*)

Fase desain adalah tahap untuk menentukan sistem yang dibangun akan berjalan dalam hal *hardware*, *software*, serta infrastruktur jaringan seperti usier *interface*, pengisian formulir dan laporan. Selain itu juga menentukan desain pada *database*, program, dan file-file yang diperlukan. Fase ini desain memiliki 4 tahap [16]:

- Pembuatan strategi desain untuk mengklarifikasikan apakah sistem akan dirancang dan dibangun oleh programmer pada perusahaan sendiri atau di-*outsource* dari perusahaan lain.
- Pembuatan desain arsitektur pada sistem untuk menjelaskan *hardware*, *software*, dan infrastruktur jaringan yang akan digunakan.
- Pembuatan *database* dan spesifikasi file atau folder untuk menentukan bagaimana data sistem akan disimpan.
- Pembuatan desain program untuk menentukan program yang akan dibuat dan apa yang akan dilakukan oleh program.

2.1.8.4. Tahapan Implementasi (*Implementation*)

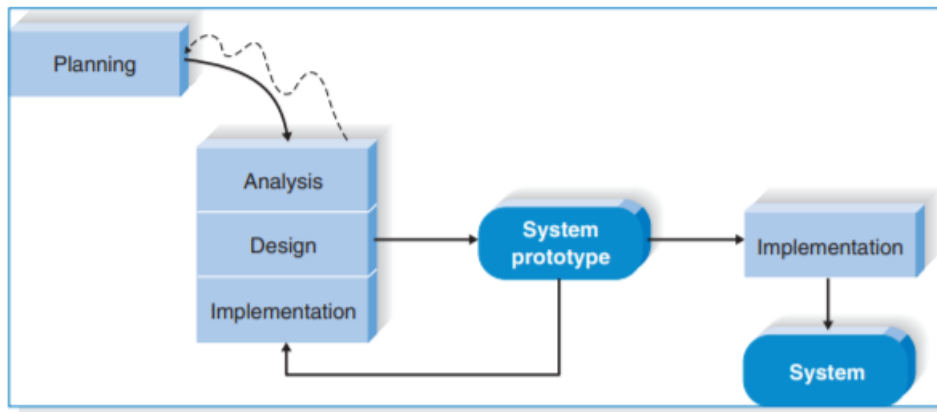
Fase implementasi merupakan fase terakhir pada SDLC, fase dimana sebuah sistem sudah dirancang dan dibuat. Fase ini memiliki 3 tahap [16]:

- Konstruksi sistem, dimana sistem akan dibangun dan diuji untuk memastikan sudah berjalan sesuai dengan apa yang direncanakan.

- Instalasi sistem, dimana sistem lama akan digantikan dengan sistem baru.
- Menetapkan rencana dukungan untuk mengidentifikasi perubahan apa saja yang diperlukan untuk sistem, maupun itu perubahan berskala besar atau kecil.

2.1.9. Metode *Prototype*

Metode *prototype* merupakan salah satu model SDLC yang digunakan dalam pengembangan sebuah *software*. Cara kerja penggunaan metode ini diawali dengan tahap *planning* untuk mencari tahu apa saja yang diperlukan dan tujuan dari pembangunan suatu sistem. Kemudian dilanjutkan dengan tahap *analysis*, *design*, dan *implementation*. Hasil yang didapatkan adalah versi sederhana dari sistem yang diajukan. Sistem tersebut kemudian diberikan kepada pengguna yang akan menguji proses kerjanya sistem tersebut. Setelah diuji, pengguna akan memberikan sebuah evaluasi dan masukan terhadap sistem tersebut. Reaksi dan komentar yang diberikan oleh pengguna akan diterima oleh para *developers*. *Developers* kemudian akan menganalisa ulang, desain ulang, dan mengimplementasikan ulang sistem *prototype* kedua yang mengoreksikan apa yang kurang dari sistem *prototype* sebelumnya. Siklus ini akan terus berjalan sampai akhirnya pihak pengguna, analis, dan sponsor menyetujui bahwa sistem *prototype* yang dibuat sudah mencukupi kebutuhan organisasi. Kelebihan dari metode ini bahwa sangat berguna jika pengguna sulit untuk menjelaskan kelayakan apa saja yang diperlukan sistem. Kekurangan dari metode ini pada biasanya terjadi karena kurangnya perhatian terhadap analisis sistem sebelum membuat keputusan desain dan implementasi [16]. Berikut merupakan gambar proses siklus pengembangan *software* metode *prototyping*.



Gambar 2.3 Proses model prototyping

Sumber: Dennis, Alan, et al., System Analysis & Design, 5th ed. United States of America : Wiley, 2012.

2.1.10. Unified Modelling Language (UML)

UML atau *Unified Modeling Language* adalah bahasa permodelan yang terdiri dari serangkaian diagram yang terintegrasi untuk membantu pengembangan sistem dan menentukan, memvisualisasikan, merancang, dan mendokumentasikan sistem yang dibangun. Diagram yang digunakan pada UML dibagi menjadi 2 kelompok pemodelan; pemodelan struktural (*Structure diagrams*) dan pemodelan perilaku (*Behaviour diagrams*) [16].

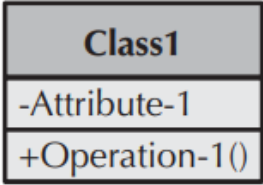

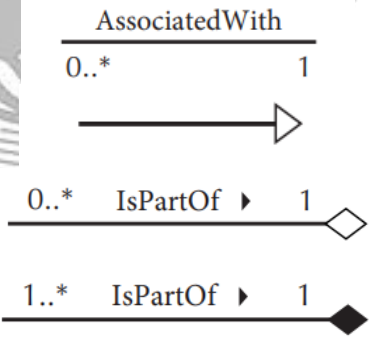
2.1.10.1. Pemodelan Struktural

Pemodelan struktural adalah pemodelan UML yang menggambarkan struktur data dari sistem yang dirancang. Tujuannya adalah membuat sebuah *vocabulary* yang bisa digunakan oleh sistem analis dan pengguna. Pemodelan ini tidak menunjukkan komponen *software* atau *class-class* pada bahasa pemrograman *object-oriented* tetapi mewakili tanggung jawab masing-masing *class* dan kolaborasi antar *class-class* tersebut. Diagram pemodelan struktural yang akan digunakan pada penelitian ini adalah *class diagram* [16].

2.1.10.1.1. Class Diagram

Class diagram adalah sebuah model statik yang menunjukkan *class-class* dan hubungan antar class yang tetap konstan dalam sistem dari waktu ke waktu. Berikut merupakan beberapa elemen yang digunakan pada pembuatan *class diagram* [16].

Tabel 2.1 Elemen-elemen pada *class diagram*

Deskripsi	Gambar
<p>Class:</p> <ul style="list-style-type: none"> - Merepresentasikan orang, tempat, barang, atau perilaku yang dimana sistem perlu untuk menyimpan data. - Memiliki daftar attribute-attribute pada bagian tengah table. - Memiliki daftar operasi-operasi pada bagian bawah table. 	 <p>The diagram shows a rectangular box representing a class. The top section is shaded and contains the text 'Class1'. The middle section contains '-Attribute-1'. The bottom section contains '+Operation-1()'.</p>
<p>Attribute:</p> <ul style="list-style-type: none"> - Merepresentasikan properties yang menjelaskan kondisi atau keadaan sebuah object. 	<p style="text-align: center;">attribute name /derived attribute name</p>
<p>Operation:</p> <ul style="list-style-type: none"> - Merepresentasikan aksi atau fungsi yang dapat dilakukan oleh sebuah class. - Bisa diklasifikasikan sebagai constructor, query, atau operasi update. - Mengandung tanda kurung yang dapat diisi dengan sebuah parameter atau informasi lain yang diperlukan untuk menjalankan operasi. 	 <p>The diagram shows a rectangular box containing the text 'operation name ()'.</p>
<p>Relationship:</p> <ul style="list-style-type: none"> - Garis yang menunjukkan relasi antara sebuah class dengan class yang lain. - Terdapat beberapa jenis relasi seperti association, generalization, aggregation, composition. 	 <p>The diagram shows three types of relationships between two classes:</p> <ul style="list-style-type: none"> 1. Association: A solid line with an open arrowhead. Multiplicity '0..*' on the left and '1' on the right. 2. Aggregation: A solid line with an open diamond at the end. Multiplicity '0..*' on the left and '1' on the right. The text 'IsPartOf' is written above the line. 3. Composition: A solid line with a filled diamond at the end. Multiplicity '1..*' on the left and '1' on the right. The text 'IsPartOf' is written above the line.

Sumber: Dennis, Alan, et al., System Analysis & Design, 5th ed. United States of America : Wiley, 2015.

2.1.10.2. Pemodelan Perilaku

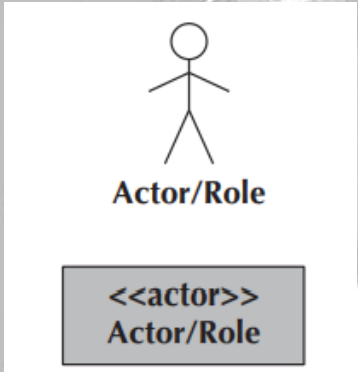

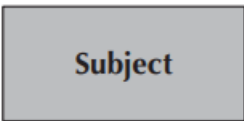
Pemodelan perilaku menggambarkan aspek dinamis bagian internal sistem yang dirancang. Tujuan utama dari pemodelan ini adalah objek-objek yang mendasari dalam domain masalah akan bekerja sama untuk membentuk sebuah kolaborasi yang

mendukung setiap *use case* pada sistem. Sementara pemodelan struktural menunjukkan hubungan antar objek, pemodelan perilaku menunjukkan pandangan internal dari proses bisnis yang telah dijelaskan oleh *use case*. Diagram pemodelan perilaku yang akan digunakan pada penelitian ini adalah *use case diagram* dan *activity diagram* [16].

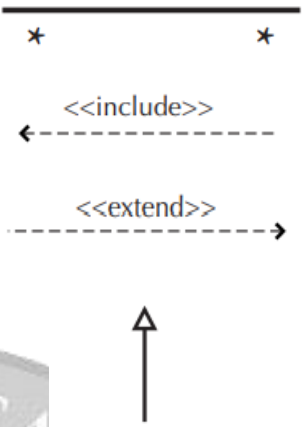
2.1.10.2.1. Use Case Diagram

Use case diagram adalah pemodelan diagram yang menunjukkan apa yang akan dilakukan sistem dan bagaimana sistem akan berjalan dengan sederhana. Diagram ini juga membantu menyederhanakan penjelasan mengenai cara kerja sistem kepada pengguna dan menunjukkan berbagai jenis *user* yang akan berinteraksi dengan sistem tersebut. Berikut merupakan beberapa elemen yang digunakan pada pembuatan *use case diagram* [16].

Tabel 2.2 Elemen-elemen pada *use case diagram*

Deskripsi	Gambar
<p>Actor:</p> <ul style="list-style-type: none"> - Merepresentasikan orang atau sistem yang memperoleh manfaat dari dan berada di luar subjek. - Ditampilkan sebagai stickman jika menunjukkan actor manusia. - Ditampilkan sebagai kotak dengan kata <<actor>> jika menunjukkan actor benda mati. - Diletakkan di luar subject boundary. 	
<p>Use Case:</p> <ul style="list-style-type: none"> - Merepresentasikan fungsi-fungsi sebuah sistem. - Dapat di-extend atau di-include ke use case lain. - Ditempatkan di dalam subject boundary. - Diberikan label kata kerja deskriptif. 	
<p>Subject Boundary:</p> <p>Merepresentasikan ruang lingkup suatu subject maupun itu sebuah sistem secara keseluruhan atau bisnis proses individual.</p>	

Tabel 2.2 Elemen-elemen pada *use case diagram*


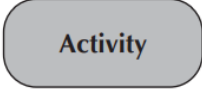
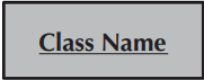

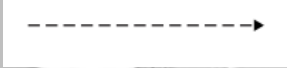



Deskripsi	Gambar
<p>Relationship:</p> <ul style="list-style-type: none"> - Sebagai penghubung antar elemen dengan fungsinya masing-masing. - Association relationship menghubungkan actor dengan use case. - Include dan extend relationship menghubungkan use case dengan use case lainnya. - Generalization relationship menghubungkan use case khusus dengan use case lain yang lebih umum. 	

Sumber: Dennis, Alan, et al., *System Analysis & Design*, 5th ed. United States of America : Wiley, 2015.


2.1.10.2.2. *Activity Diagram*

Activity diagram adalah pemodelan diagram yang digunakan untuk menunjukkan bisnis proses pada tiap objek secara individual. Diagram ini bisa digunakan untuk memodelkan alur kerja bisnis yang melibatkan banyak *use case* dan juga memodelkan alur kerja tiap masing-masing *use case*. Berikut merupakan beberapa elemen yang digunakan pada pembuatan *activity diagram* [16].

Tabel 2.3 Elemen-elemen pada *activity diagram*

Deskripsi	Gambar
<p><i>Action:</i></p> <ul style="list-style-type: none"> - Menunjukkan perilaku yang sederhana 	
<p><i>Activity:</i></p> <ul style="list-style-type: none"> - Menunjukkan suatu tindakan 	
<p><i>Object Node:</i></p> <ul style="list-style-type: none"> - Menunjukkan suatu objek yang terhubung dengan objek <i>flow</i> 	
<p><i>Control Flow:</i></p> <ul style="list-style-type: none"> - Menunjukkan pengurutan tindakan yang dilakukan 	
<p><i>Object Flow:</i></p> <ul style="list-style-type: none"> - Menunjukkan aliran suatu objek dari satu <i>activity</i> ke <i>activity</i> yang lain 	
<p><i>Initial Node:</i></p> <ul style="list-style-type: none"> - Menunjukkan awal mula dari suatu tindakan 	
<p><i>Final-Activity Node:</i></p> <ul style="list-style-type: none"> - Menunjukkan pengakhiran dari semua tindakan pada sebuah <i>activity</i> 	
<p><i>Decision Node:</i></p> <ul style="list-style-type: none"> - Menunjukkan aliran yang terpisah dengan masing-masing jalur merupakan kondisi yang berbeda 	

Tabel 2.3 Elemen-elemen pada *activity diagram*

Deskripsi	Gambar
<p><i>Merge Node:</i></p> <ul style="list-style-type: none"> - Menunjukkan penghubungan dari dua aliran <i>activity</i> yang berbeda menjadi satu 	

Sumber: Dennis, Alan, et al., *System Analysis & Design*, 5th ed. United States of America : Wiley, 2015.

2.1.11. Pengujian UML

Dalam pengujian sebuah sistem akan adanya menghadapi beberapa *error* atau kesalahan fungsi sistem yang dirancang. Sebelum diberikan kepada pelanggan atau *user*, tentu perlu melakukan pengujian pada sistem. Pada penelitian ini, pengujian akan dilakukan dengan metode *black box testing*. *Black-box testing* adalah sebuah metode pengujian *software* yang memeriksa fungsi kerja sebuah sistem tanpa perlu mengetahui cara kerja internalnya. Penguji memberikan *input* dan mengamati *output* yang dihasilkan oleh sistem yang diuji. Dari situ penguji tinggal mengidentifikasi respon yang diberikan oleh sistem, waktu respon, masalah *interface*, masalah kinerja, masalah *usability*, dan masalah *reliability* [17].

Kelebihan dari menggunakan metode ini adalah penguji tidak perlu memiliki pengetahuan dalam mengenai *programming* atau kemampuan TI, tidak perlu mengetahui detail pengimplementasian sistem, pengujian dapat dilakukan oleh penguji *outsourc*e, kemungkinan *false positive* yang rendah, dan pengujian memiliki kompleksitas yang rendah sebab hanya memodelkan perilaku user umum. Kekurangan dari menggunakan metode ini adalah sulit untuk diautomatisasikan dan jika sebuah pengujian gagal maka akan sulit untuk mengetahui apa inti masalahnya.

2.2. Penelitian Terdahulu

Tabel 2.4 menunjukkan penelitian terdahulu yang terkandung penelitian-penelitian yang sudah pernah dilakukan. Penelitian akan digunakan sebagai bahan pengajian.

Tabel 2.4 Penelitian terdahulu

No.	Nama Peneliti	Tahun	Judul Penelitian	Hasil Penelitian
1.	Zilzi Zade Tazkia	2019	<i>Perancangan Sistem Informasi Akuntansi Laporan Keuangan Laba Rugi pada Restoran Eatboss Dengan Menggunakan PHP dan MySQL</i>	Proses pencatatan pada restoran yang diteliti masih memiliki error pada pencatatan makanan dan keuangan. Sistem informasi yang telah dibangun oleh peneliti dapat memudahkan penyusunan laporan laba rugi sehingga sesuai dengan standar akuntansi [18].
2.	Syaidina Nurfi	2020	<i>Sistem Informasi Inventori Barang Pada CV. Putra Karya Baja Dengan Metode Waterfall</i>	Pencatatan inventori produk pada perusahaan yang diteliti masih dilakukan secara manual. Sistem informasi yang dibangun telah mempermudah pengelolaan keluar masuknya barang serta pendataan pelanggan [19].
3.	Heri Yanto, Firdaus	2022	<i>Sistem Informasi Administrasi Cafe & Resto Berbasis Web</i>	Proses pemesanan pada restoran yang diteliti masih dilakukan dengan menggunakan kertas dan pen. Sistem yang dibuat oleh peneliti telah mempermudah proses pemesanan. Selain itu sistem menyediakan fungsi administrasi, dimana sistem menampilkan laporan transaksi penjualan / pemasukan [20].

Sumber: Dibuat oleh peneliti, 2021.

BAB III

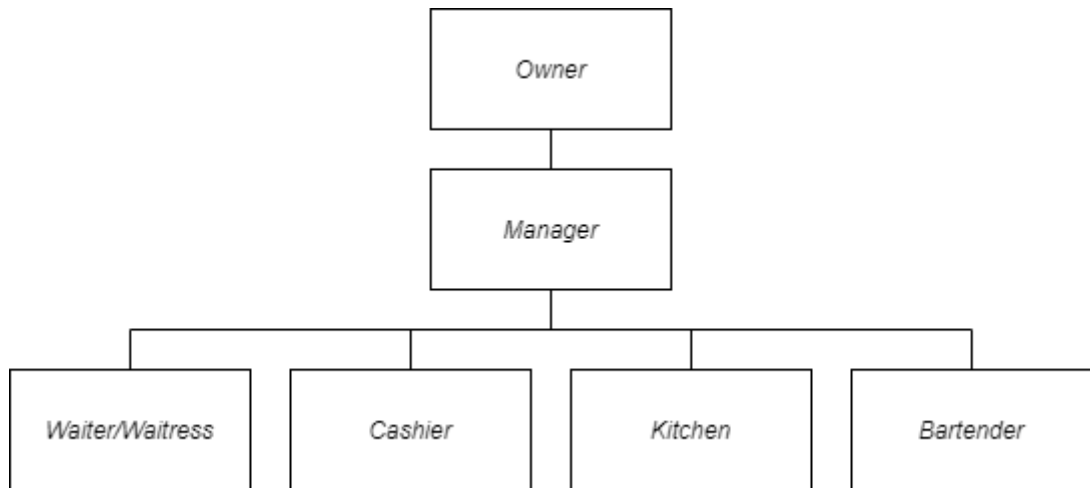
SISTEM SAAT INI

3.1. Profil Perusahaan

Awal mula berdirinya usaha Restoran Ta Chia dimulai oleh seorang pengusaha muda. Saat pemilik Restoran Ta Chia masih menduduki posisi anak bawahan perusahaan lain, ia mendapatkan sebuah keinginan untuk membangun usaha sendiri untuk memenuhi kebutuhan keluarganya. Pada saat itu, ia menemukan sebuah peluang untuk menjalankan usaha penjualan makanan. Peluang tersebut didapatkan dari makanan yang disajikan oleh keluarganya. Pada saat awal mula berjalannya usaha, proses pekerjaan masih ditangani oleh keluarganya dan beberapa karyawan. Setelah mendapatkan sebuah keuntungan yang cukup, ia mulai membuka satu cabang. Saat itu pekerjaan kasir dan pemesanan makanan masih dilakukan oleh pemilik Restoran Ta Chia. Beberapa tahun kemudian, setelah pemilik toko sudah memiliki pembelajaran, pengalaman, dan pegawai kerja yang cukup, ia memindahkan usahanya ke lokasi yang berbeda. Setelah memindahkan lokasi usaha, Restoran Ta Chia telah berhasil berjalan dengan lancar sampai dengan waktu ini. Restoran Ta Chia adalah sebuah restoran non halal yang menjual masakan bertema Chinese food. Proses pemesanan makanan oleh pelanggan masih dilakukan dengan manual. Begitu juga dengan pencatatan data seperti keuangan laba rugi, stok bahan baku dan peralatan, serta pembelian bahan baku pada *supplier*. Sampai sekarang, pemilik Restoran Ta Chia masih mencari inovasi yang baru untuk mengembangkan usahanya agar lebih maju.

3.2. Struktur Organisasi

Struktur organisasi pada usaha Restoran Ta Chia dapat dilihat pada gambar di bawah:



Gambar 3.1 Struktur Organisasi Usaha Restoran Ta Chia

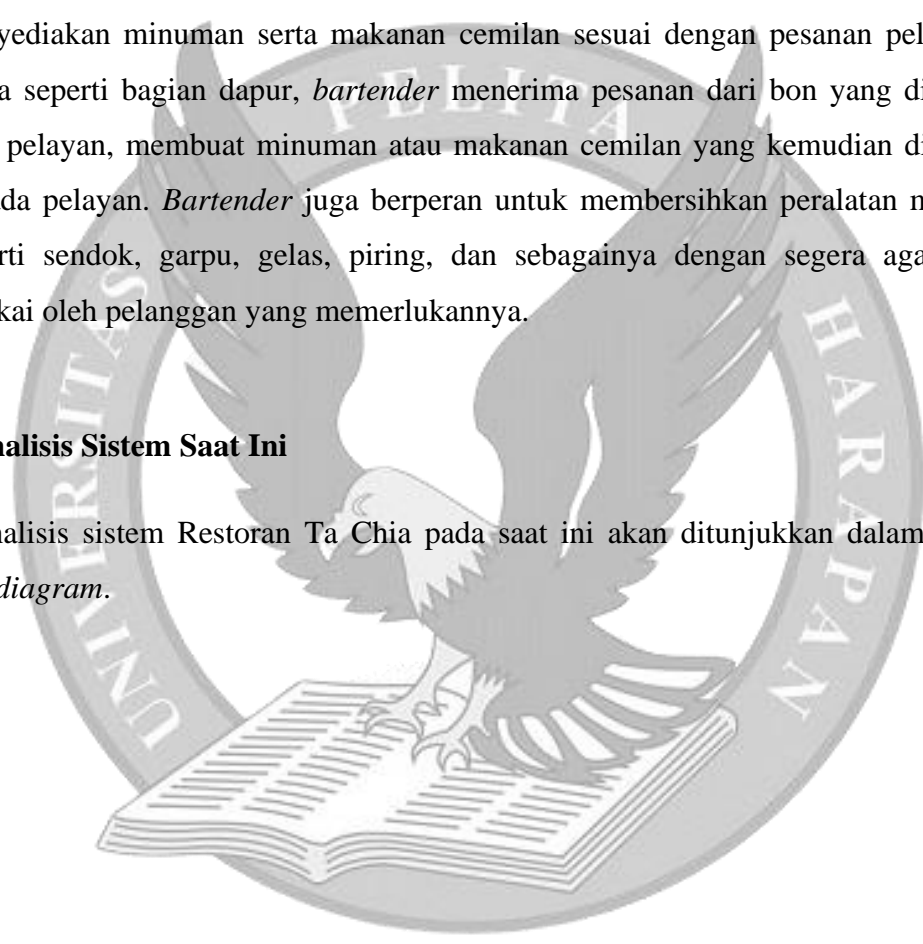
- 1) *Owner* : Berperan sebagai pemilik usaha pada Restoran Ta Chia. *Owner* memikirkan inovasi dan bisnis strategi apa saja yang baru dan dapat digunakan untuk menambahkan profit usaha. Pemilik juga memeriksa pencatatan laba rugi dan inventori, mengurusin karyawan, dan juga memantau proses kerja usahanya sehari-hari.
- 2) *Manager* : Berperan sebagai pendukung *owner* dalam mengatur dan memantau pekerjaan para karyawannya, serta menjaga keseimbangan manajemen dengan mengelola dan mengawasi pekerjaan usaha. *Manager* pada Restoran Ta Chia juga berperan dalam mengelola persediaan stok, dimana jika terjadinya kekurangan bahan baku maka *manager* akan melakukan pembelian stok kepada *supplier*. *Manager* juga berperan dalam membuat laporan stok persediaan inventori dan laporan transaksi penjualan, yang dimana laporan tersebut akan diperiksa ulang dan disimpan oleh *owner*.
- 3) *Waiter / Waitress* : Berperan dalam pelayan untuk melayani pelanggan. *Waiter / waitress* bertugas untuk menyambut pelanggan, mengambil pesanan pelanggan yang kemudian diberikan kepada *cashier*, menyajikan hidangan kepada pelanggan, dan memenuhi permintaan atau kebutuhan pelanggan dalam restoran. Makanan atau minuman yang diberikan ke pelanggan diambil dari pihak *kitchen* atau *bartender*.
- 4) *Cashier* : Berperan sebagai pengurus bagian kasir. *Cashier* bertugas untuk melayani proses transaksi pemesanan pelanggan; menerima pesanan pelanggan dari pelayan,

menghasilkan bon pemesanan pelanggan, dimana bon tersebut akan diberikan oleh *waiter / waitress* kepada pihak dapur untuk dimasak. *Cashier* juga berperan dalam melakukan pengumpulan bon transaksi penjualan per hari dan diberikan kepada *manager*, dimana ia akan menggunakannya dalam membuat laporan.

- 5) *Kitchen* : Berperan sebagai pengurus bagian dapur. *Kitchen* menerima pesanan dari pelayan dan membuat makanan sesuai dengan pesanan yang diterima. Makanan yang dibuat diberikan kepada pihak *bartender*, dimana pelayan akan menyajikan makanannya kepada pelanggan.
- 6) *Bartender* : Memiliki peran yang serupa dengan peran bagian dapur. *Bartender* menyediakan minuman serta makanan cemilan sesuai dengan pesanan pelanggan. Sama seperti bagian dapur, *bartender* menerima pesanan dari bon yang diberikan oleh pelayan, membuat minuman atau makanan cemilan yang kemudian diberikan kepada pelayan. *Bartender* juga berperan untuk membersihkan peralatan makanan seperti sendok, garpu, gelas, piring, dan sebagainya dengan segera agar dapat dipakai oleh pelanggan yang memerlukannya.

3.3. Analisis Sistem Saat Ini

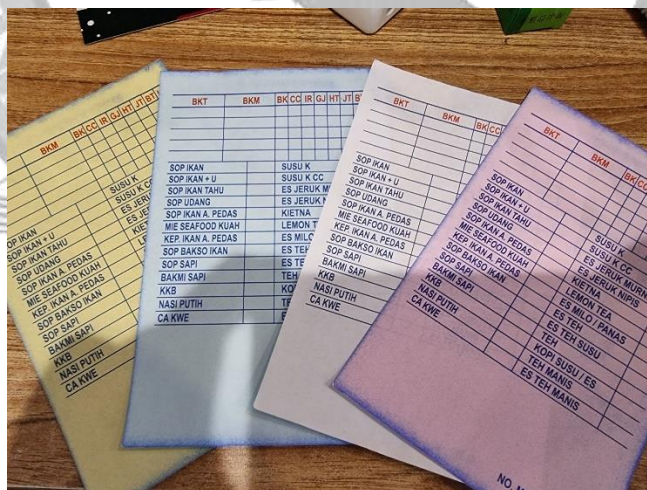
Analisis sistem Restoran Ta Chia pada saat ini akan ditunjukkan dalam bentuk *activity diagram*.



Berikut adalah langkah-langkah proses pemesanan pelanggan pada Restoran Ta Chia:

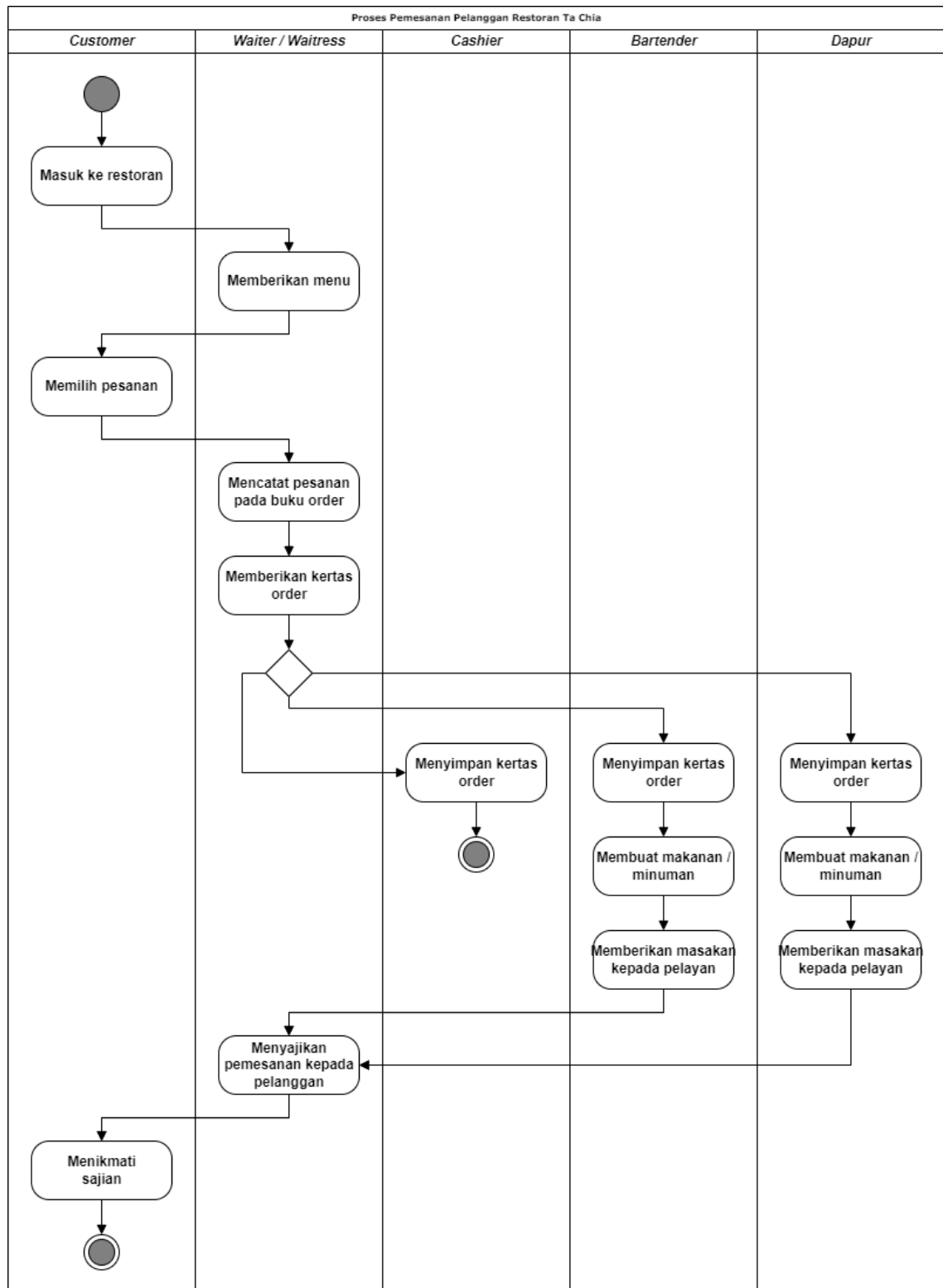
- 1) Pelanggan datang langsung ke dalam restoran dan mencari tempat duduk.
- 2) Pelayan memberikan menu makanan / minuman kepada pelanggan.
- 3) Pelanggan melakukan pemesanan.
- 4) Pelayan mencatat pesanan yang diinginkan pelanggan pada buku order.
- 5) Pelayan memberikan kertas order (ungu, kuning, putih) kepada bagian kasir, bartender, dan dapur, sedangkan pelayan menyimpan kertas order warna biru untuk melakukan pengecekan saat menyajikan makanan / minuman.
- 6) Kasir menerima kemudian menyimpan kertas order (warna ungu).
- 7) Bartender menerima kertas order (warna kuning) dan membuat makanan/minuman sesuai pemesanan.
- 8) Dapur menerima kertas order (warna putih) dan membuat makanan/minuman sesuai pemesanan.
- 9) Bartender dan Dapur memberikan masakan kepada pelayan.
- 10) Pelayan menyajikan pemesanannya kepada pelanggan.
- 11) Pelanggan menikmati hidangannya.

Berikut merupakan gambar dari buku orderan Restoran Ta Chia:



Gambar 3.2 Kertas Order Restoran Ta Chia

Gambar berikut menunjukkan *activity diagram* dari proses pemesanan pelanggan pada Restoran Ta Chia.



Gambar 3.3 Activity Diagram Proses Pemesanan Pelanggan Restoran Ta Chia

Berikut adalah langkah-langkah proses pembayaran pada Restoran Ta Chia:

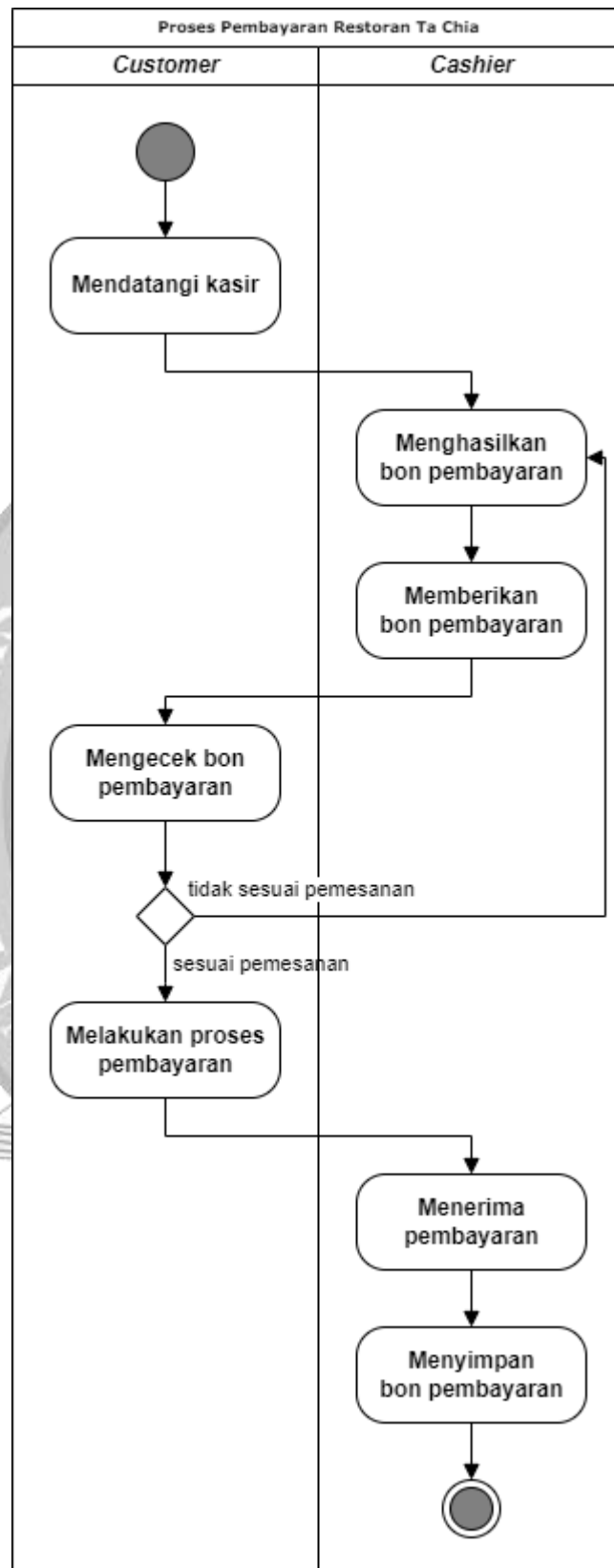
- 1) Pelanggan mendatangi kasir untuk melakukan transaksi.
- 2) Kasir menghasilkan bon pembayaran.
- 3) Kasir kemudian memberikan bon pembayaran kepada pelanggan.
- 4) Pelanggan mengecek bon pembayaran apakah sesuai dengan pemesanan.
- 5) Jika tidak sesuai dengan pemesanan maka kasir akan membuat ulang bon pembayaran. Jika sesuai dengan pemesanan maka pelanggan lanjut melakukan proses pembayaran.
- 6) Kasir menerima pembayaran pelanggan.
- 7) Kasir kemudian menyimpan bon pembayaran untuk keperluan pencatatan pemasukan.

Berikut merupakan gambar dari bon pembayaran Restoran Ta Chia:



Gambar 3.4 Bon Pembayaran Restoran Ta Chia

Gambar berikut menunjukkan *activity diagram* dari proses pembayaran pada Restoran Ta Chia.



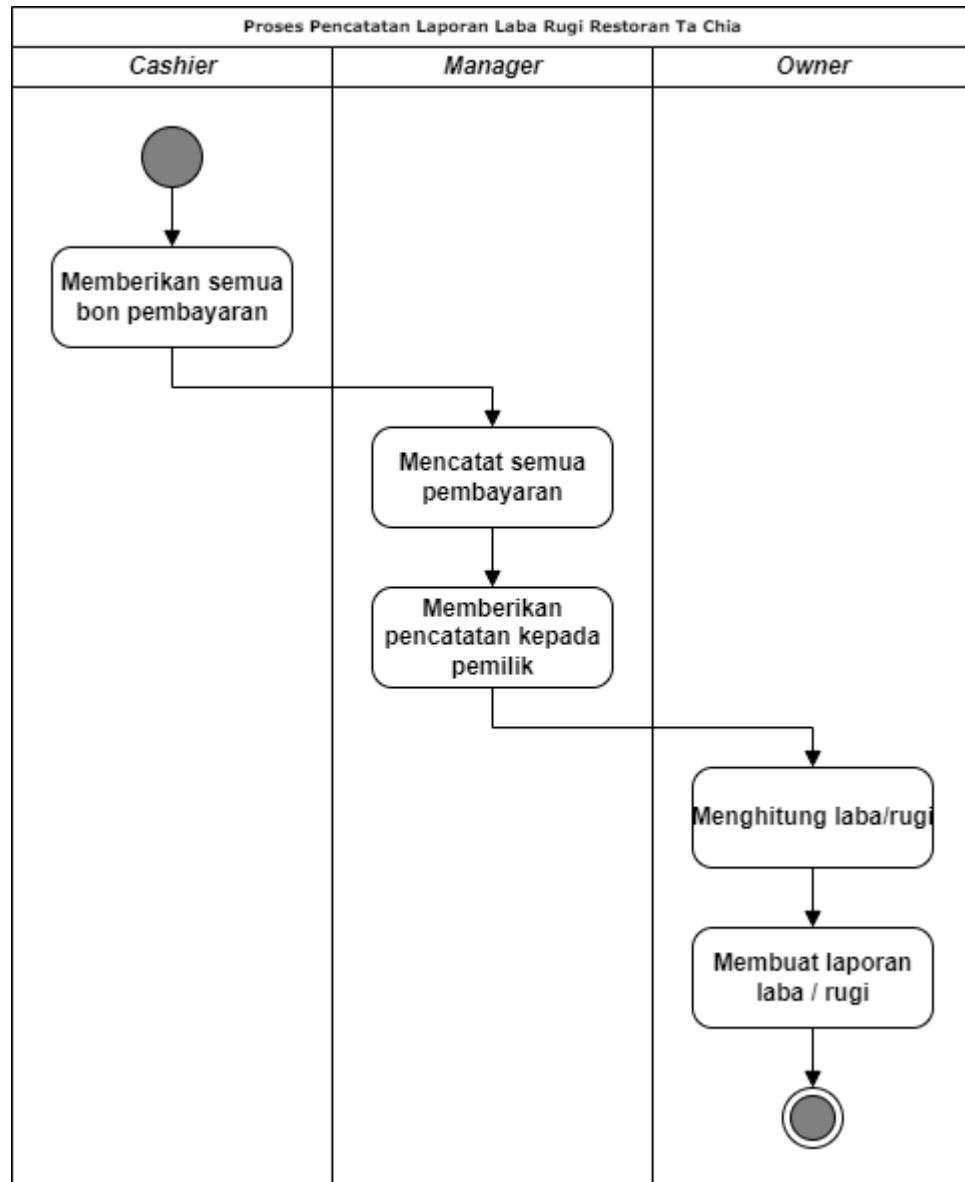
Gambar 3.5 Activity Diagram Proses Pembayaran Restoran Ta Chia

Berikut adalah langkah-langkah proses pencatatan laporan jual beli pada Restoran Ta Chia:

- 1) Kasir memberikan semua bon pembayaran yang sudah dikumpulkan kepada *manager*.
- 2) *Manager* mencatat semua pembayaran ke dalam buku laporan pemasukan.
- 3) *Manager* kemudian memberikan buku laporan pemasukan kepada pemilik.
- 4) Pemilik menghitung laba / rugi berdasarkan data pemasukan pada buku laporan.
- 5) Pemilik kemudian membuat laporan laba / rugi.



Gambar berikut menunjukkan *activity diagram* dari proses pencatatan laporan jual beli pada Restoran Ta Chia.



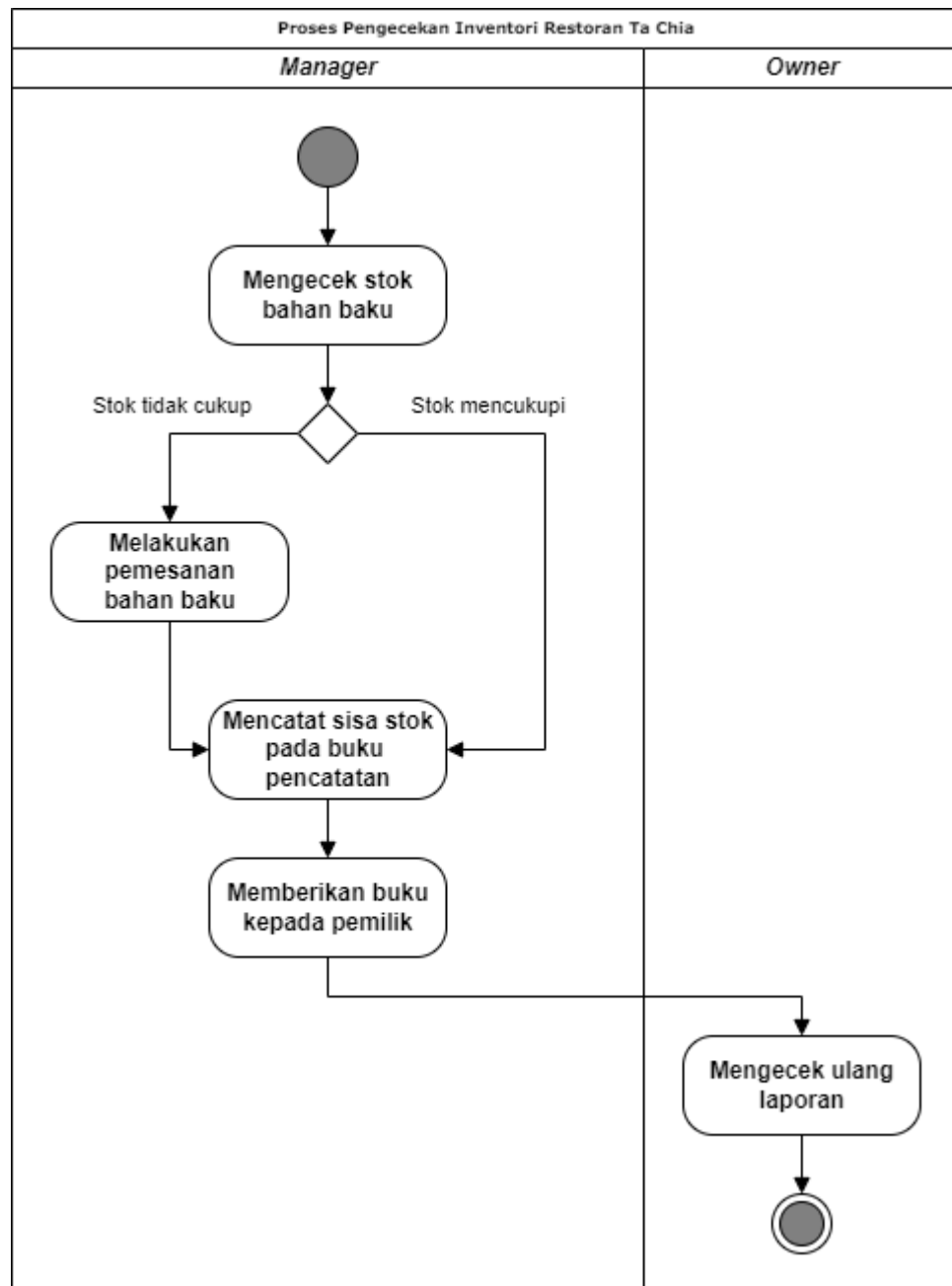
Gambar 3.6 *Activity Diagram* Proses Pencatatan Laporan Laba Rugi Restoran Ta Chia

Berikut adalah langkah-langkah proses pengecekan inventori pada Restoran Ta Chia:

- 1) *Manager* mengecek sisa stok bahan baku.
- 2) Jika stok bahan baku tidak mencukupi maka *manager* akan melakukan pemesanan bahan baku. Jika stok bahan baku mencukup maka *manager* akan lanjut mencatat sisa stok bahan baku ke dalam buku pencatatan.
- 3) *Manager* kemudian memberikan buku pencatatan kepada pemilik.
- 4) Pemilik kemudian memeriksa ulang laporan stok bahan baku.



Gambar berikut menunjukkan *activity diagram* dari proses pengecekan inventori pada Restoran Ta Chia.

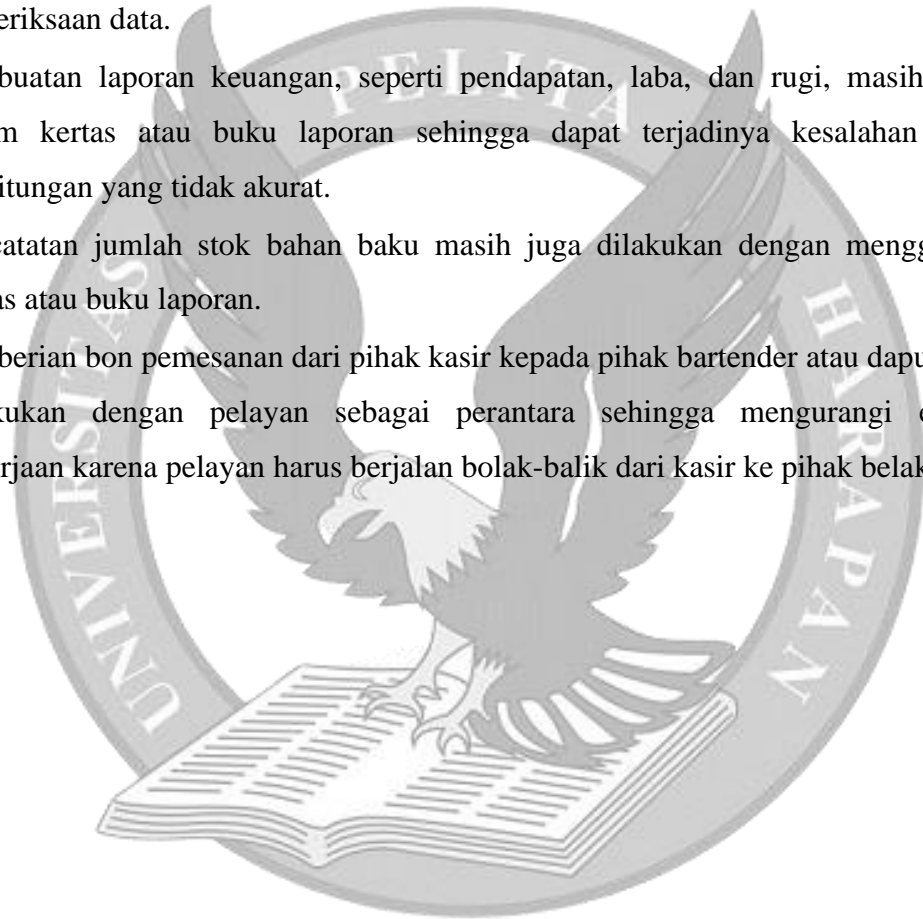


Gambar 3.7 Activity Diagram Proses Pengecekan Inventori Restoran Ta Chia

3.3.1. Kendala Sistem Saat Ini

Pengolahan/pemrosesan dari beberapa data sudah menggunakan komputer dalam bentuk Excel, tetapi data yang tersimpan belum terisi dengan rapih atau lengkap. Selain itu, data yang tersimpan ke dalam komputer belum mencakup data stok barang ataupun keuangan. Berdasarkan dari kekurangan tersebut, kendala sistem Restoran Ta Chia pada saat ini adalah sebagai berikut:

- Penyimpanan data transaksi penjualan masih disimpan dalam bentuk bon kertas yang dibuat dengan mesin kasir sehingga dapat menyulitkan pencarian dan pemeriksaan data.
- Pembuatan laporan keuangan, seperti pendapatan, laba, dan rugi, masih dicatat dalam kertas atau buku laporan sehingga dapat terjadinya kesalahan seperti perhitungan yang tidak akurat.
- Pencatatan jumlah stok bahan baku masih juga dilakukan dengan menggunakan kertas atau buku laporan.
- Pemberian bon pemesanan dari pihak kasir kepada pihak bartender atau dapur masih dilakukan dengan pelayan sebagai perantara sehingga mengurangi efisiensi pekerjaan karena pelayan harus berjalan bolak-balik dari kasir ke pihak belakang.



BAB IV

ANALISIS DAN PERANCANGAN SISTEM USULAN

4.1. Analisis Kelayakan Sistem

Sistem yang dibuat akan dilakukan analisis kelayakan sistem untuk melihat dan mengidentifikasi risiko-risiko pada sistem yang diusul. Analisis yang akan dilakukan adalah kelayakan teknis, kelayakan organisasi, dan kelayakan ekonomi [16].

4.1.1. Kelayakan Teknis (*Technical Feasibility*)

Analisis kelayakan teknis dilakukan untuk mengukur sejauh mana sistem yang akan dirancang dapat berhasil dibuat, dikembangkan, dan digunakan oleh *user*. Berikut merupakan kelayakan teknis pada sistem yang dirancang:

- 1) Risiko pengenalan aplikasi *web* oleh *user* adalah sedang. Sistem yang dibuat didesain sedemikian rupa agar *user* dapat menggunakan dan mengerti alur penggunaan *website* ini dengan mudah.
- 2) Risiko pengenalan teknologi oleh *user* adalah tinggi. Sistem yang akan dibangun bertujuan untuk membantu proses kerja bisnis yang belum mengintegrasikan penggunaan teknologi pada sistem mereka. Begitu juga dengan para karyawannya yang sudah bekerja tanpa bantuan teknologi pada proses pekerjaan mereka pada sehari-hari. Karena itu, risiko pengenalan teknologi kepada *user* adalah tinggi.
- 3) Risiko pengembangan sistem adalah tinggi. Sistem yang dibuat hanya dirancang untuk memenuhi kebutuhan usaha, dimana fitur-fitur yang tercantum adalah fungsi yang sudah ada dan biasa digunakan pada aplikasi usaha lain. Dengan demikian, perancangan sistem tidak terlalu sulit untuk dilakukan. Meskipun demikian, batas waktu dan jumlah minimal bimbingan yang diberikan selama pengembangan sistem ini dapat membuat moral perancang melemah sehingga memperlambat efisiensi kerja.

4.1.2. Kelayakan Organisasi (*Organizational Feasibility*)

Analisis kelayakan organisasi dilakukan untuk mengidentifikasi risiko yang mungkin akan dihadapi dan melihat besarnya tingkat penerimaan terhadap aplikasi oleh pihak pengguna. Sistem yang dibangun baik digunakan untuk mengelola data keuangan dan inventaris pada usaha restoran, terutama dengan usaha yang belum mengimplementasikan teknologi pada bisnisnya. Sistem yang dibangun telah didukung

dengan *feedback* dan survei dari para pekerja dan manajer pihak restoran sehingga fungsi yang akan dibuat tidak menjauh dari sistem yang mereka masih terapkan pada saat ini sehingga dapat lebih mudah digunakan. Pada rencana kedepannya, sistem ini diterapkan pada usaha bisnis mereka sebagai komputerisasian kepada proses kerja usaha.

4.1.3. Kelayakan Ekonomi (*Economic Feasibility*)

Analisis kelayakan ekonomi dilakukan untuk mengidentifikasi risiko finansial pada perancangan sistem. Rencana dari pembangunan proyek ini adalah membantu meringankan beban dan mempermudah pencatatan inventori dan keuangan restoran. Untuk menggunakan sistem usulan, diperlukannya sebuah perangkat yang digunakan sebagai server utama sistem. Berikut adalah spesifikasi minimum pada perangkat *software* dan *hardware* yang dijadikan sebagai server utama.

Tabel 4.1 Spesifikasi Perangkat Minimum Server Utama

<i>Hardware</i>	
<i>Processor</i>	Intel Core i3
RAM	8 GB
<i>Hard disk space</i>	30 GB
<i>Software</i>	
Sistem operasi	Windows 7
<i>Database</i>	MariaDB 10.4.17
<i>Web server</i>	Apache 2.4.46
<i>Browser</i>	Google Chrome

Pada saat pengembangan sistem, perangkat yang digunakan di restoran sudah memiliki spesifikasi yang cukup untuk mengembangkan dan menggunakan sistem usulan sehingga tidak perlu mengeluarkan biaya apapun. Berikut adalah spesifikasi perangkat *software* dan *hardware* yang sudah dimiliki oleh Restoran Ta Chia.

Tabel 4.2 Spesifikasi Perangkat Milik Restoran Ta Chia

Hardware		
	PC-1	PC-2
<i>Processor</i>	Inter Core i3	Intel Core i7
RAM	8 GB	12 GB
<i>Hard disk space</i>	100 GB	30 GB
Software		
	PC-1	PC-2
Sistem operasi	Windows 10	Windows 10
<i>Browser</i>	Google Chrome	Google Chrome

Pengguna sistem usulan bertujuan untuk memiliki 3 pihak atau jenis pengguna, maka dari itu diperlukan 3 perangkat yang dapat digunakan untuk menggunakan sistem usulan. Satu untuk pihak manajer, satu untuk pihak kasir, dan satu untuk pihak dapur. Restoran Ta Chia sudah memiliki perangkat yang dapat digunakan untuk menggunakan sistem usulan, termasuk perangkat pembangun sistem, sehingga tidak perlu mengeluarkan biaya tambahan untuk membeli perangkat baru. PC-1 untuk pihak manajer, PC-2 untuk pihak dapur, dan perangkat pembangun sistem sebagai server utama sistem serta digunakan untuk pihak kasir.

Sistem yang dibangun hanya akan digunakan secara lokal dan hanya boleh diakses oleh pihak restoran dan bukan pelanggan. Penghubungan perangkat dilakukan dengan koneksi alamat IP pada masing-masing perangkat dengan menggunakan *router* sebagai perantara. Maka dari itu, penggunaan sistem tidak memerlukan jasa *hosting*. Berdasarkan dari analisis ekonomi tersebut, sistem yang dirancang telah dianggap layak untuk dilaksanakan.

4.2. Tahap Analisis

Pada bagian analisis terdapat 3 tahap dari pengumpulan kebutuhan pengguna (*User Requirement Gathering*), pemodelan perilaku (*Behavior Diagram*), dan pemodelan struktural (*Structural Diagram*).

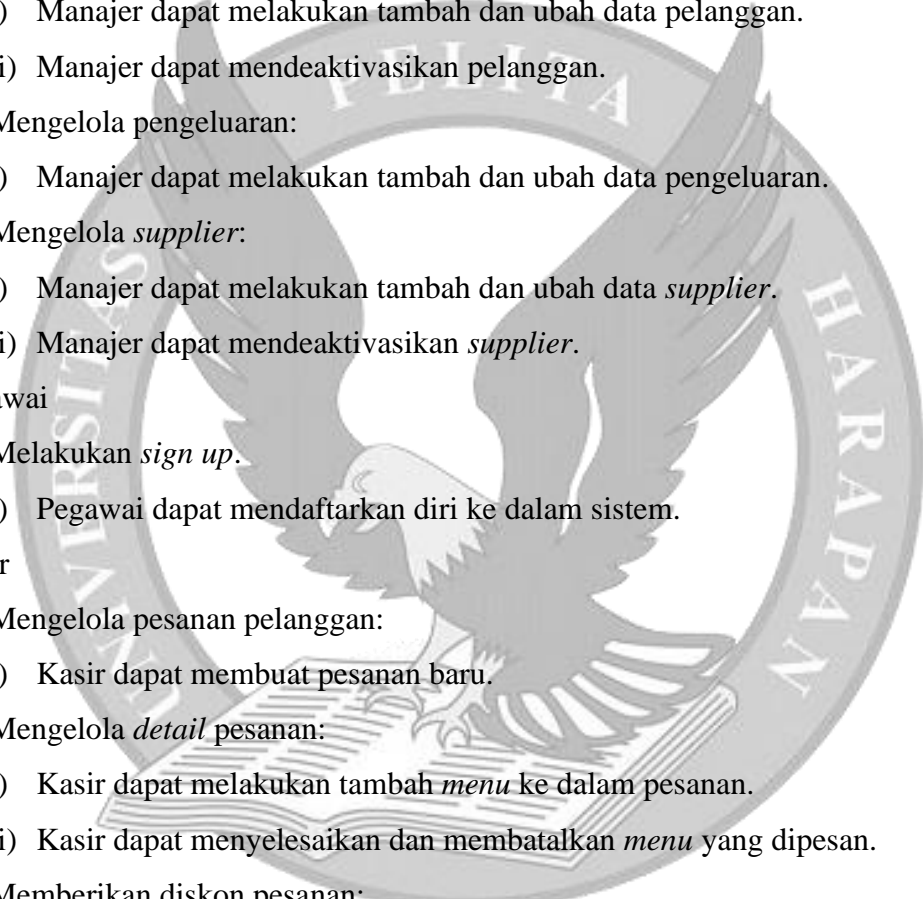
4.2.1. Pengumpulan Kebutuhan Pengguna (*User Requirement Gathering*)

Tahap pengumpulan kebutuhan pengguna, dibagikan menjadi 2 kategori yaitu kebutuhan fungsional (*functional requirement*) dan kebutuhan non-fungsional (*non-functional requirement*).

4.2.1.1. *Functional Requirement*

Kebutuhan pengguna pada penggunaan aplikasi yang dirancang dibagi menjadi 2 yaitu manajer dan pegawai, dimana pegawai dibagi menjadi pihak kasir dan pihak dapur. Semua jenis pengguna dapat melakukan *log in* sebelum menggunakan fungsi-fungsi lainnya. Berikut adalah kebutuhan fungsional pada sistem aplikasi pengelolaan keuangan dan inventoris:

- 1) Manajer
 - a) Mengelola *user*:
 - i) Manajer dapat melakukan tambah dan ubah data *user*.
 - ii) Manajer dapat mendeaktivasi *user*.
 - b) Mengelola *role*:
 - i) Manajer dapat melakukan tambah dan ubah data *role user*.
 - ii) Manajer dapat mendeaktivasi *role user*.
 - c) Mengelola *menu* makanan:
 - i) Manajer dapat melakukan tambah dan ubah data *menu* makanan.
 - ii) Manajer dapat mengaktifasi dan mendeaktivasi *menu* makanan.
 - d) Mengelola *menu detail* makanan:
 - i) Manajer dapat melakukan tambah dan ubah data *menu detail* makanan.
 - ii) Manajer dapat menghapus *menu detail* makanan pada *menu* makanan.
 - e) Mengelola kategori makanan:
 - i) Manajer dapat melakukan tambah dan ubah data kategori makanan.
 - ii) Manajer dapat mendeaktivasi kategori makanan.
 - f) Mengelola inventori barang:
 - i) Manajer dapat melakukan tambah dan ubah data inventori barang.
 - ii) Manajer dapat mengaktifasi dan mendeaktivasi inventori barang.
 - g) Menambah jumlah stok inventori barang:
 - i) Manajer dapat melakukan *add* jumlah stok barang.
 - h) Mengelola jenis barang:
 - i) Manajer dapat melakukan tambah dan ubah data jenis barang.
 - ii) Manajer dapat mendeaktivasi jenis barang.
 - i) Mengelola satuan barang:
 - i) Manajer dapat melakukan tambah dan ubah data satuan barang.
 - ii) Manajer dapat mendeaktivasi satuan barang.

- 
- j) Mengelola pemasukan:
 - i) Manajer dapat melihat data laporan pemasukan/pemesanan.
 - k) Mengelola meja:
 - i) Manajer dapat melakukan tambah dan ubah data meja.
 - ii) Manajer dapat mendeaktivasikan meja.
 - l) Mengelola metode pembayaran:
 - i) Manajer dapat melakukan tambah dan ubah data metode pembayaran.
 - ii) Manajer dapat men-deaktivasikan metode pembayaran.
 - m) Mengelola pelanggan:
 - i) Manajer dapat melakukan tambah dan ubah data pelanggan.
 - ii) Manajer dapat mendeaktivasikan pelanggan.
 - n) Mengelola pengeluaran:
 - i) Manajer dapat melakukan tambah dan ubah data pengeluaran.
 - o) Mengelola *supplier*:
 - i) Manajer dapat melakukan tambah dan ubah data *supplier*.
 - ii) Manajer dapat mendeaktivasikan *supplier*.
 - 2) Pegawai
 - a) Melakukan *sign up*.
 - i) Pegawai dapat mendaftarkan diri ke dalam sistem.
 - 3) Kasir
 - a) Mengelola pesanan pelanggan:
 - i) Kasir dapat membuat pesanan baru.
 - b) Mengelola *detail* pesanan:
 - i) Kasir dapat melakukan tambah *menu* ke dalam pesanan.
 - ii) Kasir dapat menyelesaikan dan membatalkan *menu* yang dipesan.
 - c) Memberikan diskon pesanan:
 - i) Kasir dapat memberikan diskon kepada tiap *menu*.
 - d) Bayar pesanan:
 - i) Kasir dapat melakukan proses pembayaran pesanan.
 - 4) Dapur/Bartender
 - a) Melakukan konfirmasi pesanan:
 - i) Dapur/Bartender dapat menerima dan menyelesaikan pesanan baru.
 - ii) Konfirmasi pesanan hanya dapat dilakukan setelah kasir melakukan pengelolaan *detail* pesanan, yaitu menambah *menu* baru ke dalam pesanan.

4.2.1.2. *Non-Functional Requirement*

Kebutuhan non-fungsional pada umumnya dibagi menjadi 4 kategori yaitu keamanan, operasional, reliabilitas, dan budaya. Berikut adalah kebutuhan fungsional pada sistem aplikasi pengelolaan keuangan dan inventaris:

- 1) Keamanan
 - a) Sistem mengharuskan pengguna untuk melakukan identifikasi diri melalui nama dan *password* pengguna.
 - b) Kode *password* yang tercantum telah fungsi *password_hash* pada Codeigniter 3.
 - c) Alamat IP pada perangkat telah diatur supaya hanya pihak internal yang dapat menggunakan sistem.
- 2) Operasional
 - a) Sistem diakses dengan perangkat yang terhubung dengan jaringan *wifi* yang sama.
 - b) Sistem dapat diakses secara lokal.
 - c) Sistem dapat diakses selama jam kerja (07:00 – 21:00).
- 3) Reliabilitas
 - a) Sistem dapat memberikan data pesanan secara *real time*.
- 4) Budaya
 - a) Bahasa yang digunakan pada sistem adalah bahasa Indonesia, dikarenakan sistem hanya digunakan pada usaha restoran yang bekerja aktif pada daerah lokal. Begitu juga dengan mayoritas karyawan yang lahir dan tinggal di Indonesia. Maka dari itu bahasa yang digunakan pada sistem adalah bahasa Indonesia agar dapat diterima dan dimengerti oleh para pengguna.

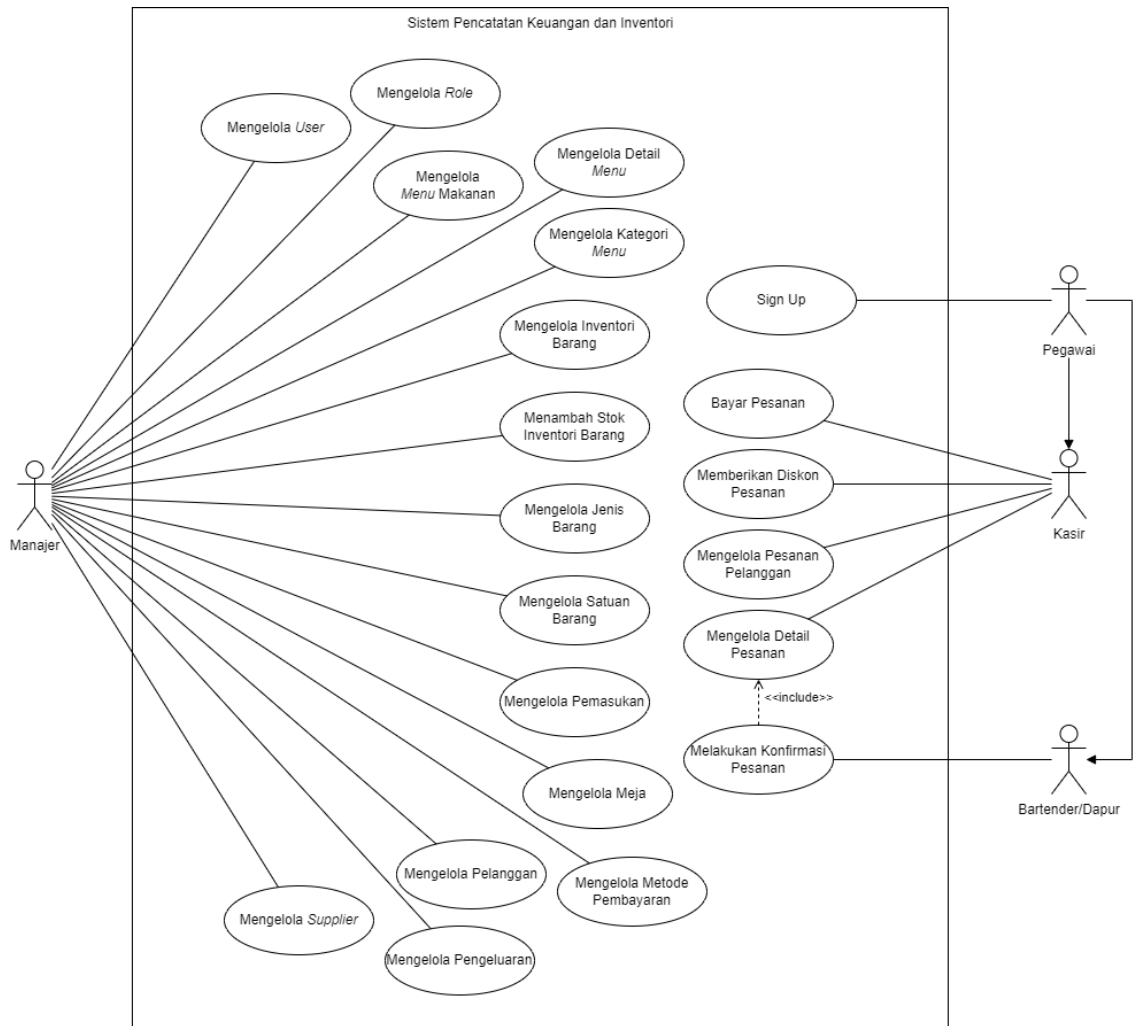
4.3. Sistem Usulan

4.3.1. Pemodelan Fungsional Pemodelan Struktural

Tahap pemodelan fungsional dibagi menjadi 2 pemodelan yaitu *use case diagram* dan *activity diagram* untuk menjelaskan proses berjalannya aktivitas pengguna pada sistem yang dibangun.

4.3.1.1. Use Case Diagram

Pembuatan *use case diagram* digunakan untuk melihat dan memahami fungsi-fungsi yang dapat dilakukan pada sistem dan aktor-aktor yang menggunakan sistem tersebut. Berikut adalah *use case diagram* pada sistem aplikasi pencatatan keuangan dan inventoris.



Gambar 4.1 Use Case Diagram Sistem Usulan

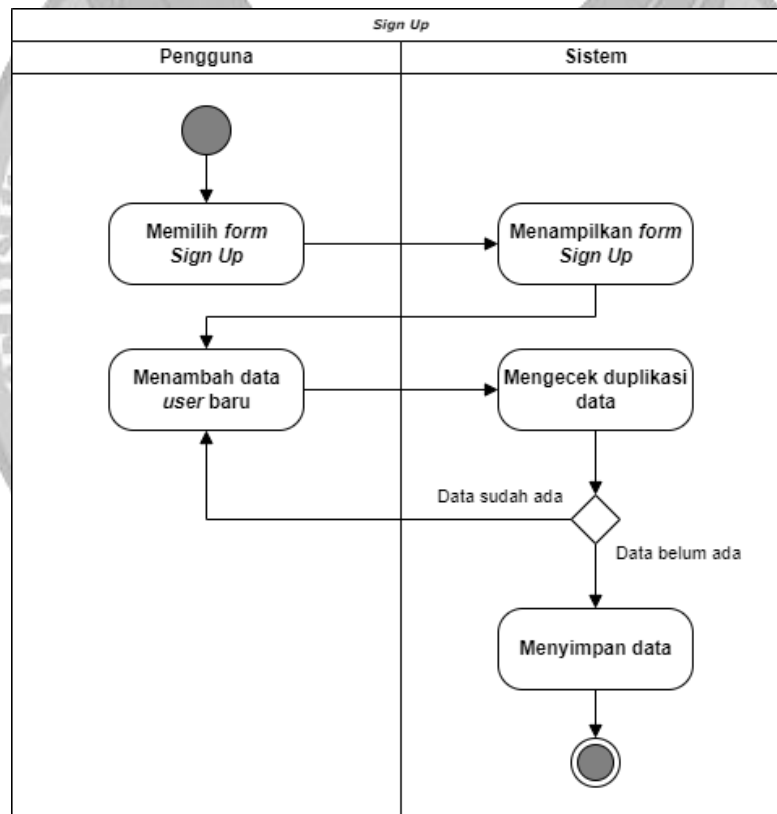
Gambar di atas adalah *use case diagram* dari sistem pencatatan keuangan dan inventori dimana *actor* pengguna sistem dibagi menjadi 4, yaitu manajer, pegawai, kasir, dan dapur/bartender. Manajer dapat mengelola semua jenis data-data yang berlaku pada usahanya seperti *user*, barang, pemasukan dan pengeluaran, dan lain-lain. Pegawai dapat melakukan *sign up*. Kasir dapat mengelola data yang berhubungan dengan pemesanan seperti menambah pesanan, mengedit pesanan, dan mencatat pembayaran. Dapur/Bartender dapat melakukan konfirmasi pesanan baru yang telah *input* oleh pihak kasir.

4.3.1.2. Activity Diagram

1) Sign Up

Berikut adalah *activity diagram* dari *Sign Up*. Langkah-langkah proses *sign up* terdiri dari berikut:

- Pengguna memilih *form Sign Up*.
- Sistem menampilkan *form Sign Up*.
- Pengguna menambah data *user* baru.
- Sistem akan mengecek duplikasi data.
- Jika data sudah ada di dalam *database*, maka sistem menampilkan *alert message* dan lanjut menampilkan *form Sign Up*.
- Jika data belum ada di dalam *database*, maka sistem akan menyimpan data *user* baru dan lanjut menampilkan *form Sign Up*.

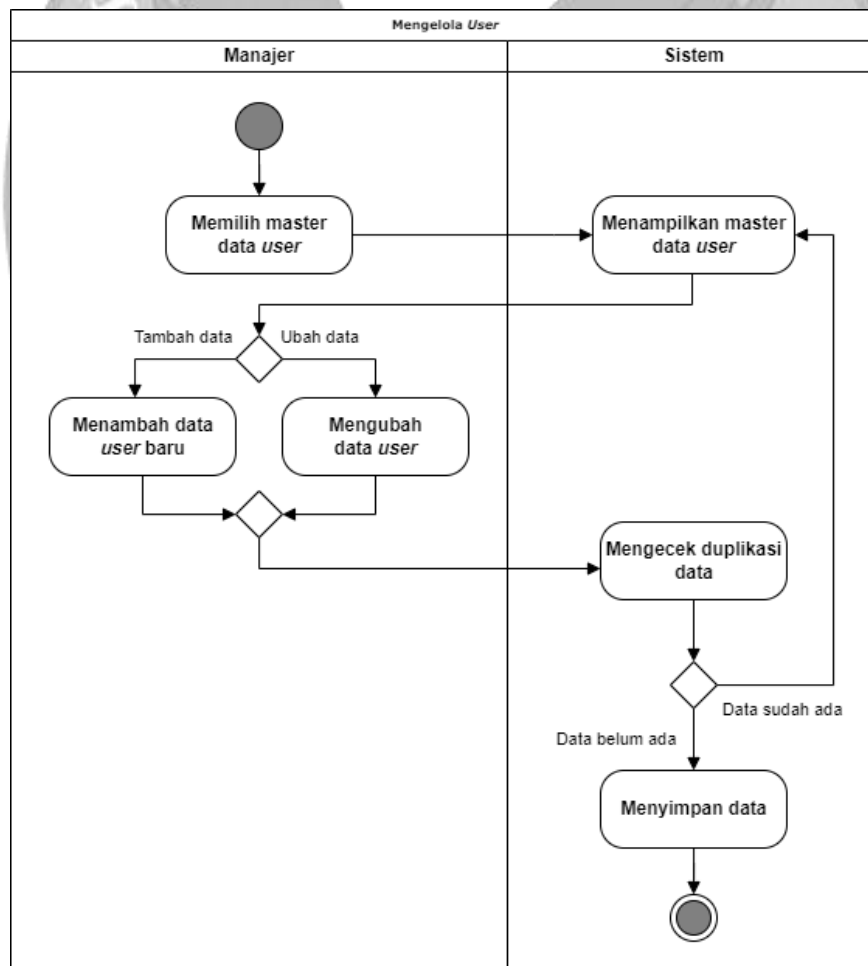


Gambar 4.2 Activity Diagram Sign Up

2) Mengelola *User*

Berikut adalah *activity diagram* dari Mengelola *User*. Langkah-langkah proses mengelola *user* terdiri dari berikut:

- a) Manajer memilih halaman *master data user*.
- b) Sistem menampilkan halaman *master data user*.
- c) Jika manajer ingin menambah data baru, maka manajer memilih *form input* data baru dan menambah data user baru.
- d) Jika manajer ingin mengubah data, maka manajer memilih *form edit* data dan mengubah data *user*.
- e) Sistem akan mengecek duplikasi data.
- f) Jika data sudah ada di dalam *database*, maka sistem menampilkan *alert message* dan balik ke halaman sebelumnya.
- g) Jika data belum ada di dalam *database*, maka sistem akan menyimpan data *user* baru dan lanjut menampilkan halaman *master data user*.

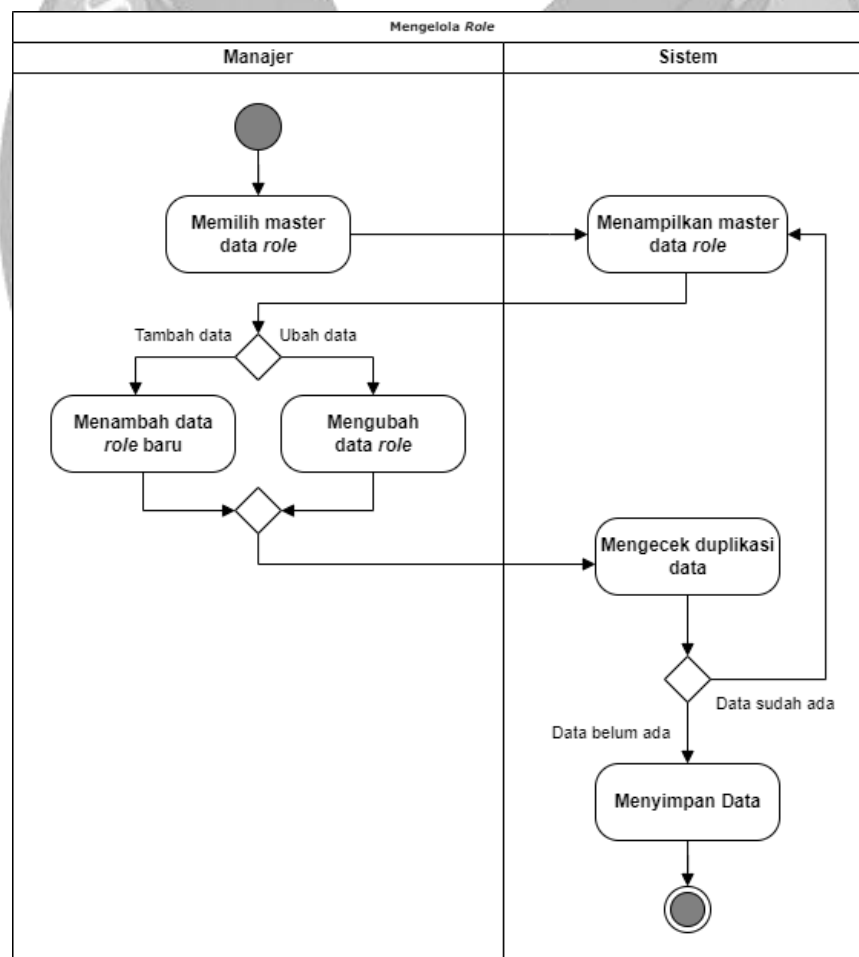


Gambar 4.3 Activity Diagram Mengelola *User*

3) Mengelola *Role*

Berikut adalah *activity diagram* dari Mengelola *Role*. Langkah-langkah proses mengelola *role* terdiri dari berikut:

- a) Manajer memilih halaman *master data role*.
- b) Sistem menampilkan halaman *master data role*.
- c) Jika manajer ingin menambah data baru, maka manajer memilih *form input* data baru dan menambah data *role* baru.
- d) Jika manajer ingin mengubah data, maka manajer memilih *form edit* data dan mengubah data *role*.
- e) Sistem akan mengecek duplikasi data.
- f) Jika data sudah ada di dalam *database*, maka sistem menampilkan *alert message* dan balik ke halaman sebelumnya.
- g) Jika data belum ada di dalam *database*, maka sistem akan menyimpan data *role* baru dan lanjut menampilkan halaman *master data role*.

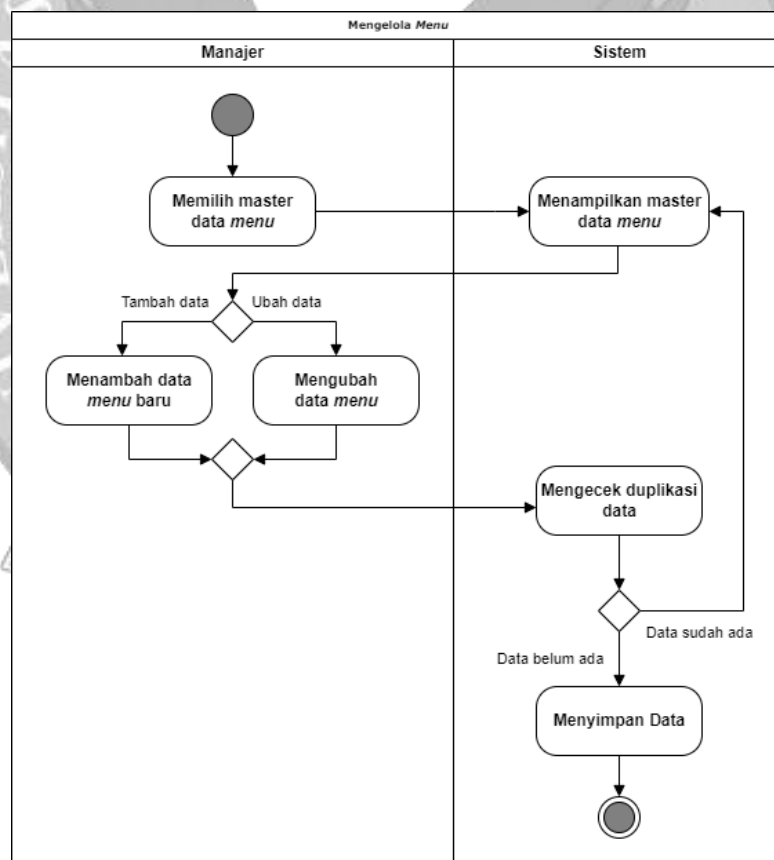


Gambar 4.4 Activity Diagram Mengelola Role

4) Mengelola *Menu*

Berikut adalah *activity diagram* dari Mengelola *Menu*. Langkah-langkah proses mengelola *menu* terdiri dari berikut:

- a) Manajer memilih halaman *master data menu*.
- b) Sistem menampilkan halaman *master data menu*.
- c) Jika manajer ingin menambah data baru, maka manajer memilih *form input* data baru dan menambah data *menu* baru.
- d) Jika manajer ingin mengubah data, maka manajer memilih *form edit* data dan mengubah data *menu*.
- e) Sistem akan mengecek duplikasi data.
- f) Jika data sudah ada di dalam *database*, maka sistem menampilkan *alert message* dan balik ke halaman sebelumnya.
- g) Jika data belum ada di dalam *database*, maka sistem akan menyimpan data *menu* baru dan lanjut menampilkan halaman *master data menu*.

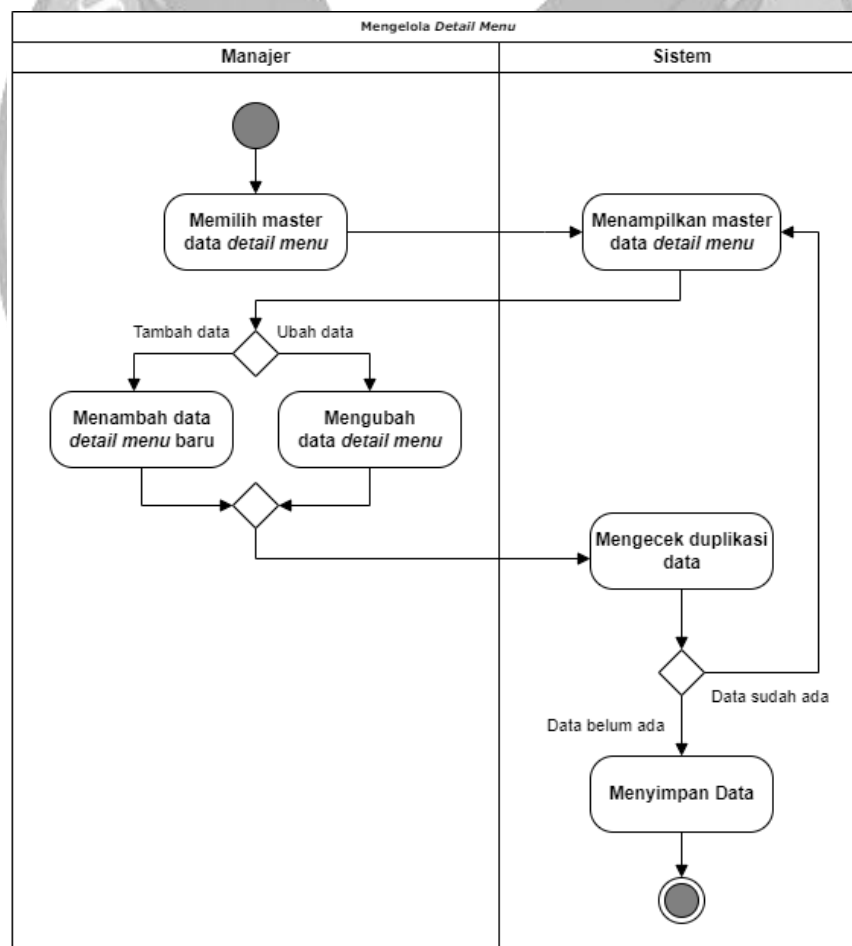


Gambar 4.5 Activity Diagram Mengelola Menu

5) Mengelola *Detail Menu*

Berikut adalah *activity diagram* dari Mengelola *Detail Menu*. Langkah-langkah proses mengelola *menu* terdiri dari berikut:

- a) Manajer memilih halaman *master data detail menu*.
- b) Sistem menampilkan halaman *master data detail menu*.
- c) Jika manajer ingin menambah data baru, maka manajer memilih *form input* data baru dan menambah data *detail menu* baru.
- d) Jika manajer ingin mengubah data, maka manajer memilih *form edit* data dan mengubah data *detail menu*.
- e) Sistem akan mengecek duplikasi data.
- f) Jika data sudah ada di dalam *database*, maka sistem menampilkan *alert message* dan balik ke halaman sebelumnya.
- g) Jika data belum ada di dalam *database*, maka sistem akan menyimpan data *detail menu* baru dan lanjut menampilkan halaman *master data detail menu*.

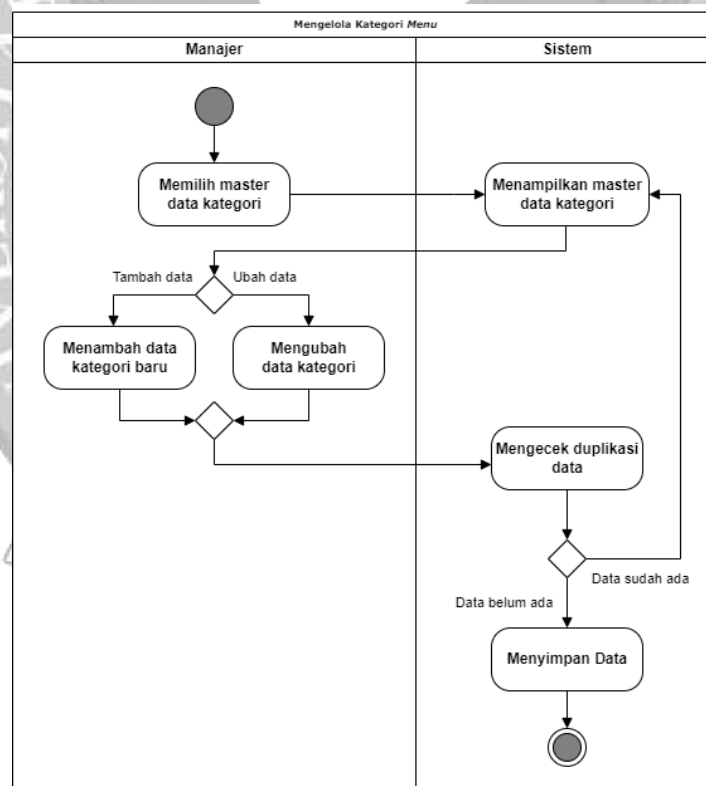


Gambar 4.6 Activity Diagram Mengelola Detail Menu

6) Mengelola Kategori *Menu*

Berikut adalah *activity diagram* dari Mengelola Kategori *Menu*. Langkah-langkah proses mengelola kategori *menu* terdiri dari berikut:

- a) Manajer memilih halaman *master data* kategori *menu*.
- b) Sistem menampilkan halaman *master data* kategori *menu*.
- c) Jika manajer ingin menambah data baru, maka manajer memilih *form input* data baru dan menambah data kategori *menu* baru.
- d) Jika manajer ingin mengubah data, maka manajer memilih *form edit* data dan mengubah data kategori *menu*.
- e) Sistem akan mengecek duplikasi data.
- f) Jika data sudah ada di dalam *database*, maka sistem menampilkan *alert message* dan balik ke halaman sebelumnya.
- g) Jika data belum ada di dalam *database*, maka sistem akan menyimpan data kategori *menu* baru dan lanjut menampilkan halaman *master data* kategori *menu*.

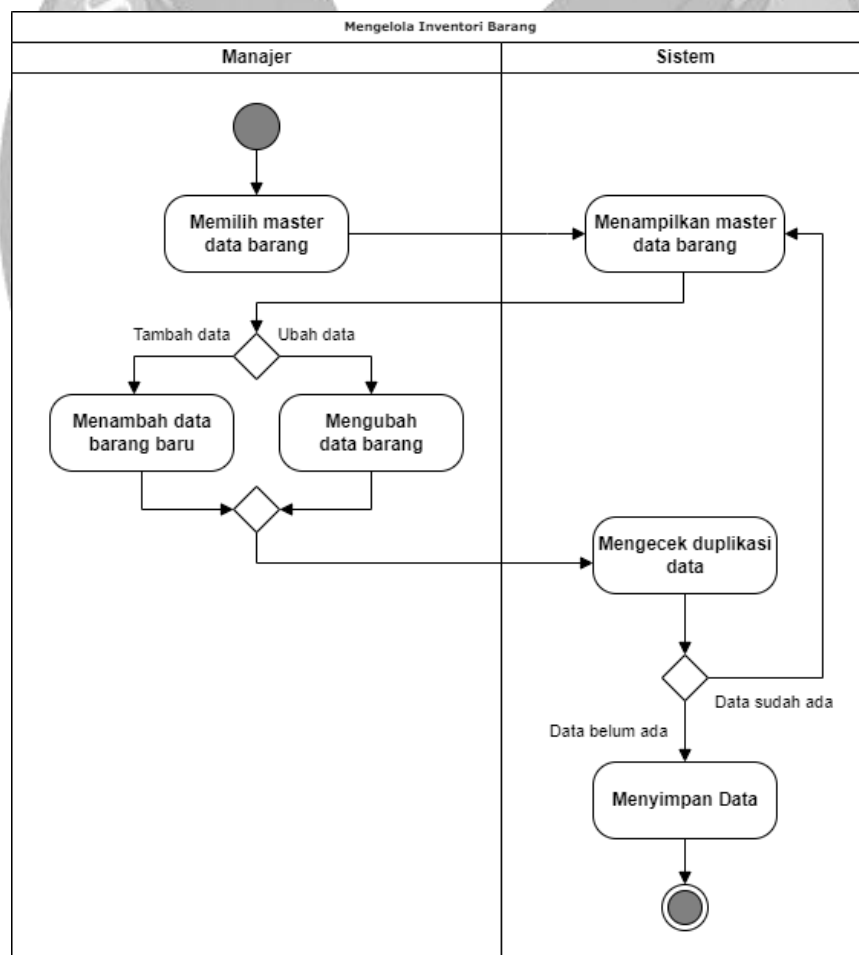


Gambar 4.7 *Activity Diagram* Mengelola Kategori *Menu*

7) Mengelola Inventori Barang

Berikut adalah *activity diagram* dari Mengelola Inventori Barang. Langkah-langkah proses mengelola inventori barang terdiri dari berikut:

- a) Manajer memilih halaman *master data* inventori barang.
- b) Sistem menampilkan halaman *master data* inventori barang.
- c) Jika manajer ingin menambah data baru, maka manajer memilih *form input* data baru dan menambah data barang baru.
- d) Jika manajer ingin mengubah data, maka manajer memilih *form edit* data dan mengubah data barang.
- e) Sistem akan mengecek duplikasi data.
- f) Jika data sudah ada di dalam *database*, maka sistem menampilkan *alert message* dan balik ke halaman sebelumnya.
- g) Jika data belum ada di dalam *database*, maka sistem akan menyimpan data barang baru dan lanjut menampilkan halaman *master data* barang.

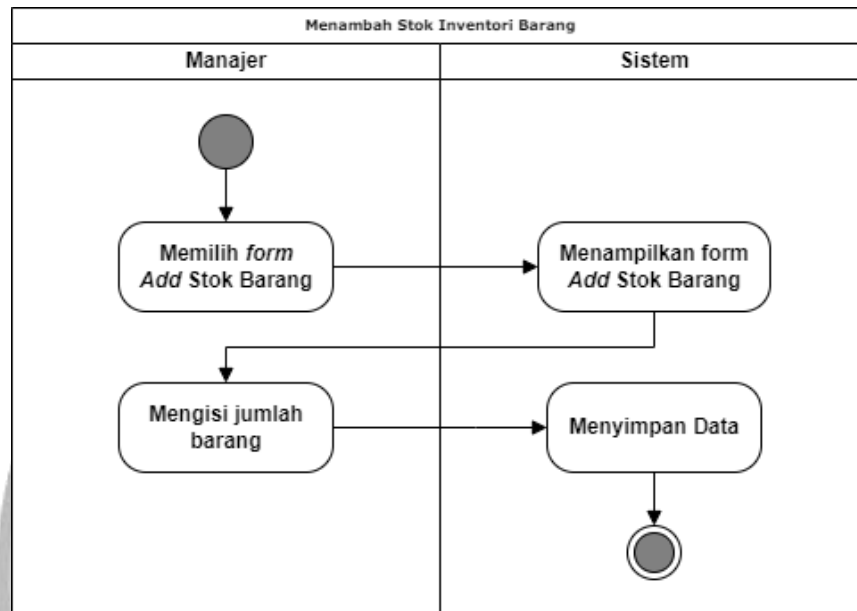


Gambar 4.8 Activity Diagram Mengelola Inventori Barang

8) Menambah Stok Inventori Barang

Berikut adalah *activity diagram* dari Menambah Stok Inventori Barang. Langkah-langkah proses menambah stok inventori barang terdiri dari berikut:

- a) Manajer memilih *form add* stok barang.
- b) Sistem menampilkan *form add* stok barang.
- c) Manajer menambah jumlah barang.
- d) Sistem akan menambahkan jumlah sisa stok dan menyimpan data di *database*.

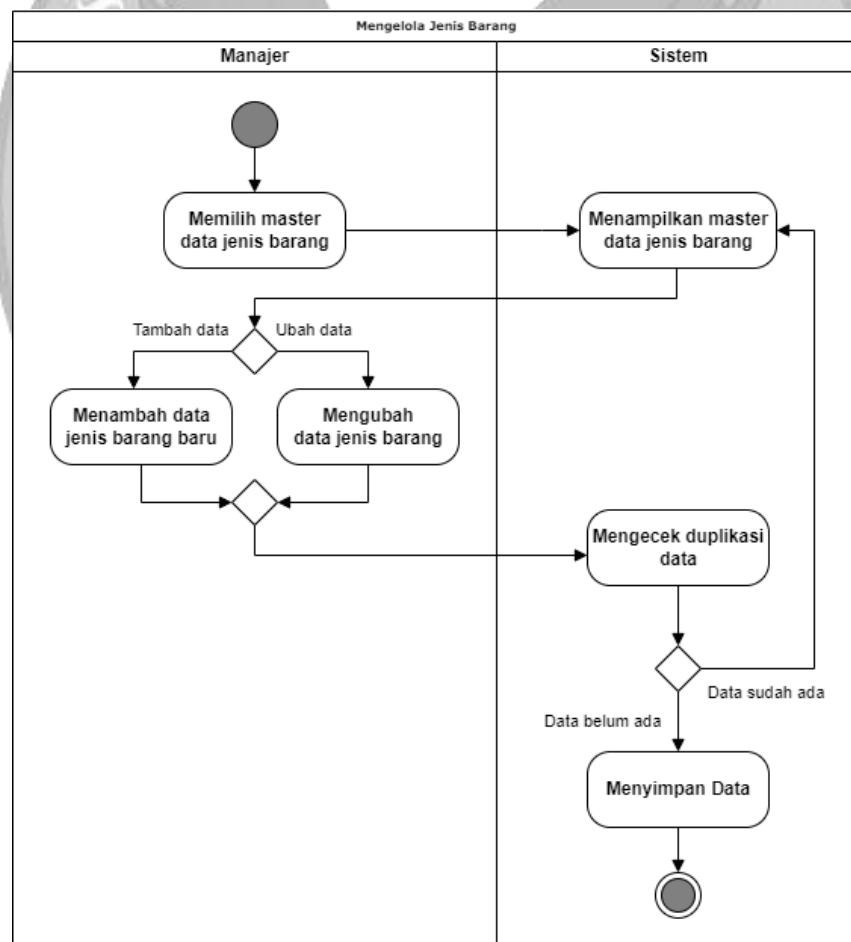


Gambar 4.9 Activity Diagram Manambah Stok Inventori Barang

9) Mengelola Jenis Barang

Berikut adalah *activity diagram* dari Mengelola Jenis Barang. Langkah-langkah proses mengelola jenis barang terdiri dari berikut:

- a) Manajer memilih halaman *master data* jenis barang.
- b) Sistem menampilkan halaman *master data* jenis barang.
- c) Jika manajer ingin menambah data baru, maka manajer memilih *form input* data baru dan menambah data jenis barang baru.
- d) Jika manajer ingin mengubah data, maka manajer memilih *form edit* data dan mengubah data jenis barang.
- e) Sistem akan mengecek duplikasi data.
- f) Jika data sudah ada di dalam *database*, maka sistem menampilkan *alert message* dan balik ke halaman sebelumnya.
- g) Jika data belum ada di dalam *database*, maka sistem akan menyimpan data jenis barang baru dan lanjut menampilkan halaman *master data* jenis barang.

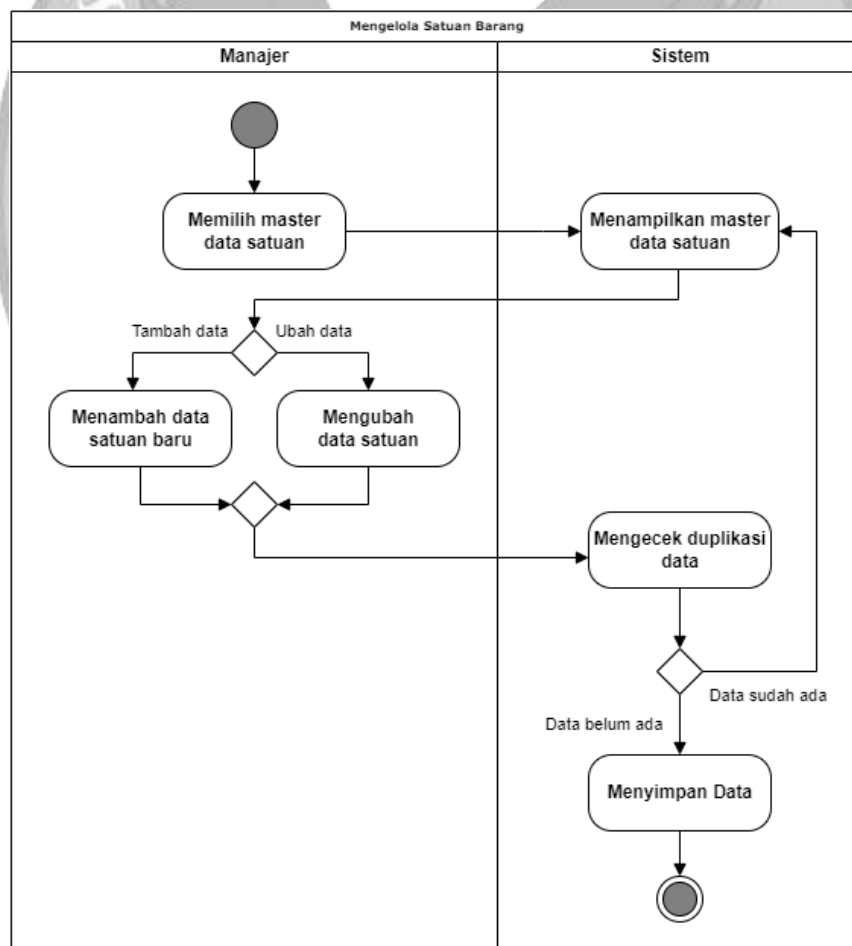


Gambar 4.10 Activity Diagram Mengelola Jenis Barang

10) Mengelola Satuan Barang

Berikut adalah *activity diagram* dari Mengelola Satuan Barang. Langkah-langkah proses mengelola satuan barang terdiri dari berikut:

- a) Manajer memilih halaman *master data* satuan barang.
- b) Sistem menampilkan halaman *master data* satuan barang.
- c) Jika manajer ingin menambah data baru, maka manajer memilih *form input* data baru dan menambah data satuan barang baru.
- d) Jika manajer ingin mengubah data, maka manajer memilih *form edit* data dan mengubah data satuan barang.
- e) Sistem akan mengecek duplikasi data.
- f) Jika data sudah ada di dalam *database*, maka sistem menampilkan *alert message* dan balik ke halaman sebelumnya.
- g) Jika data belum ada di dalam *database*, maka sistem akan menyimpan data satuan barang baru dan lanjut menampilkan halaman *master data* satuan barang.

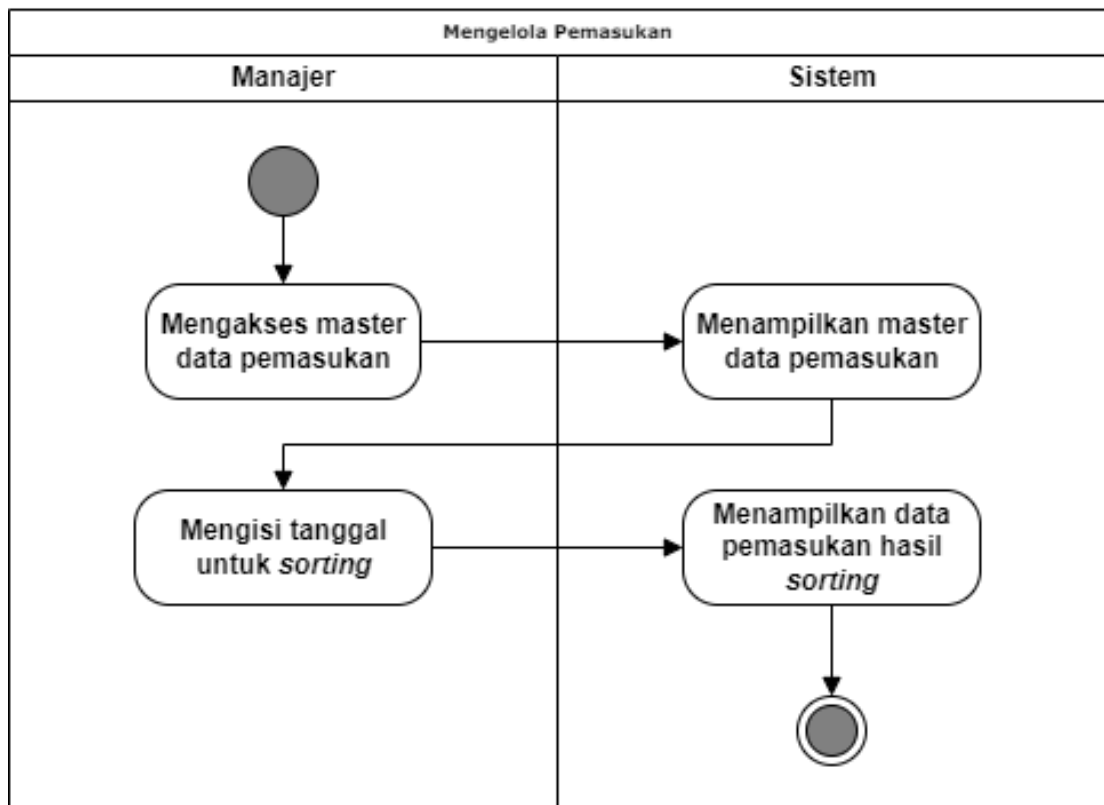


Gambar 4.11 Activity Diagram Mengelola Satuan Barang

11) Mengelola Pemasukan

Berikut adalah *activity diagram* dari Mengelola Pemasukan. Langkah-langkah proses mengelola meja terdiri dari berikut:

- Manajer memilih halaman *master data* pemasukan.
- Sistem menampilkan halaman *master data* pemasukan.
- Manajer mengisi tanggal awal dan akhir untuk melakukan *sorting* data berdasarkan tanggalnya.
- Sistem menampilkan data pemasukan hasil *sorting*.

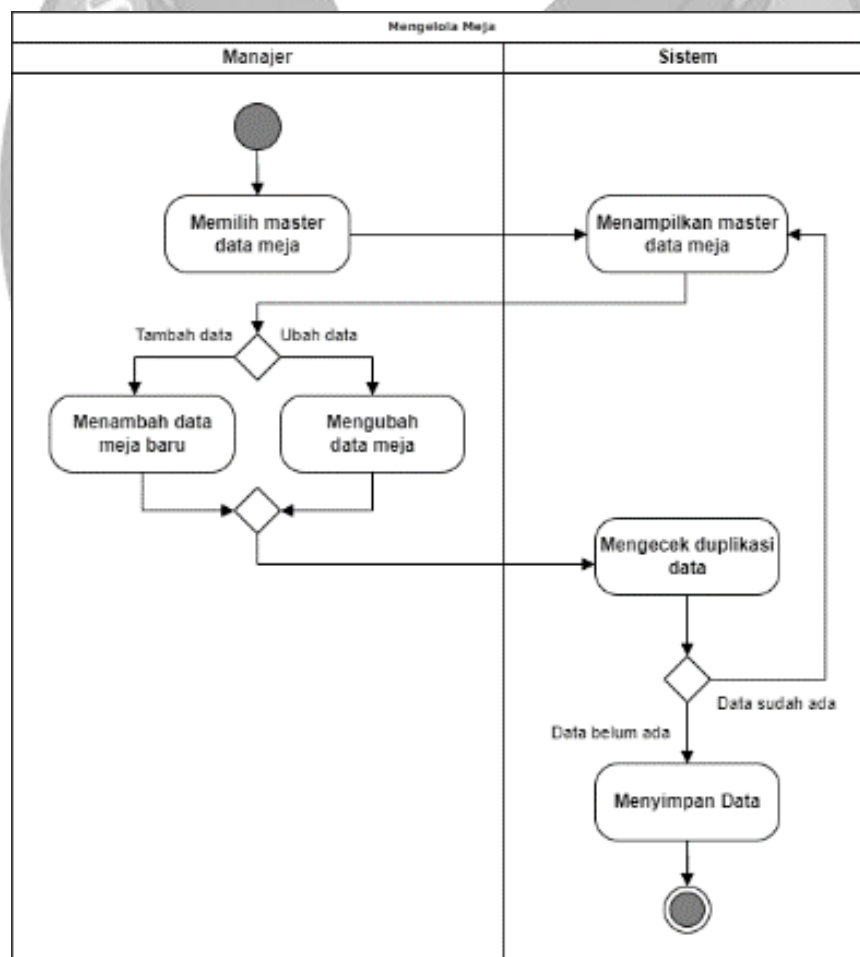


Gambar 4.12 Activity Diagram Mengelola Pemasukan

12) Mengelola Meja

Berikut adalah *activity diagram* dari Mengelola Meja. Langkah-langkah proses mengelola meja terdiri dari berikut:

- a) Manajer memilih halaman *master data* meja.
- b) Sistem menampilkan halaman *master data* meja.
- c) Jika manajer ingin menambah data baru, maka manajer memilih *form input* data baru dan menambah data meja baru.
- d) Jika manajer ingin mengubah data, maka manajer memilih *form edit* data dan mengubah data meja.
- e) Sistem akan mengecek duplikasi data.
- f) Jika data sudah ada di dalam *database*, maka sistem menampilkan *alert message* dan balik ke halaman sebelumnya.
- g) Jika data belum ada di dalam *database*, maka sistem akan menyimpan data meja baru dan lanjut menampilkan halaman *master data* meja.

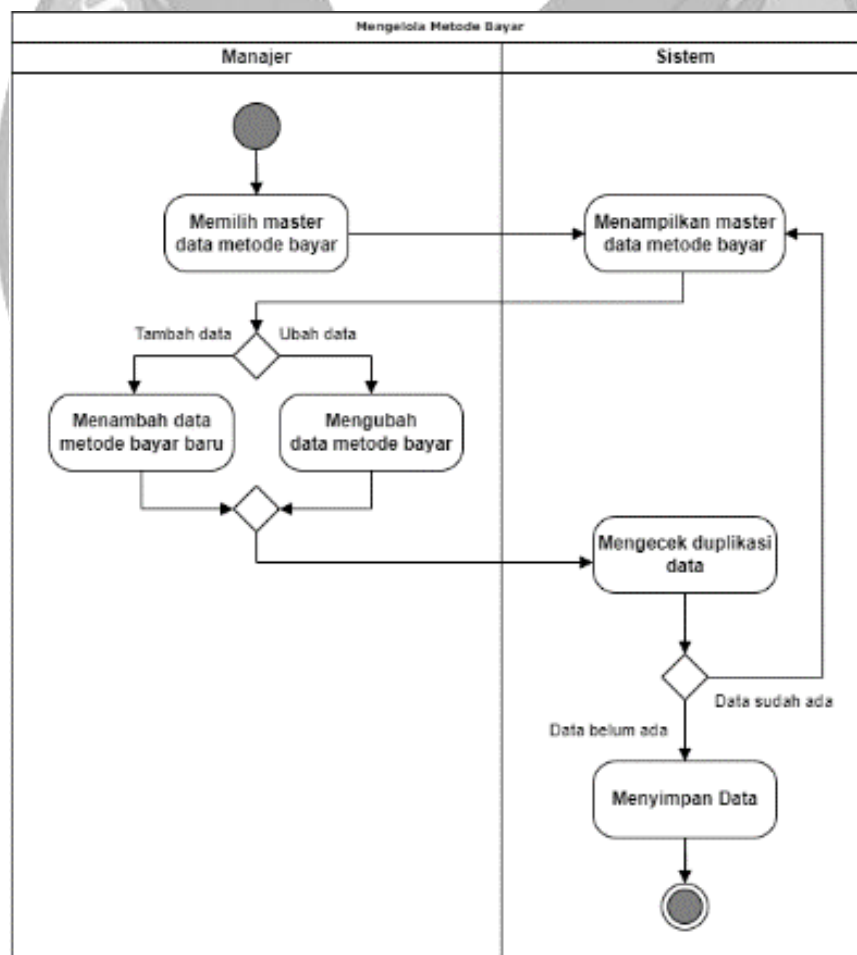


Gambar 4.13 Activity Diagram Mengelola Meja

13) Mengelola Metode Bayar

Berikut adalah *activity diagram* dari Mengelola Metode Bayar. Langkah-langkah proses mengelola metode bayar terdiri dari berikut:

- a) Manajer memilih halaman *master data* metode bayar.
- b) Sistem menampilkan halaman *master data* metode bayar.
- c) Jika manajer ingin menambah data baru, maka manajer memilih *form input* data baru dan menambah data metode bayar baru.
- d) Jika manajer ingin mengubah data, maka manajer memilih *form edit* data dan mengubah data metode bayar.
- e) Sistem akan mengecek duplikasi data.
- f) Jika data sudah ada di dalam *database*, maka sistem menampilkan *alert message* dan balik ke halaman sebelumnya.
- g) Jika data belum ada di dalam *database*, maka sistem akan menyimpan data metode bayar baru dan lanjut menampilkan halaman *master data* metode bayar.

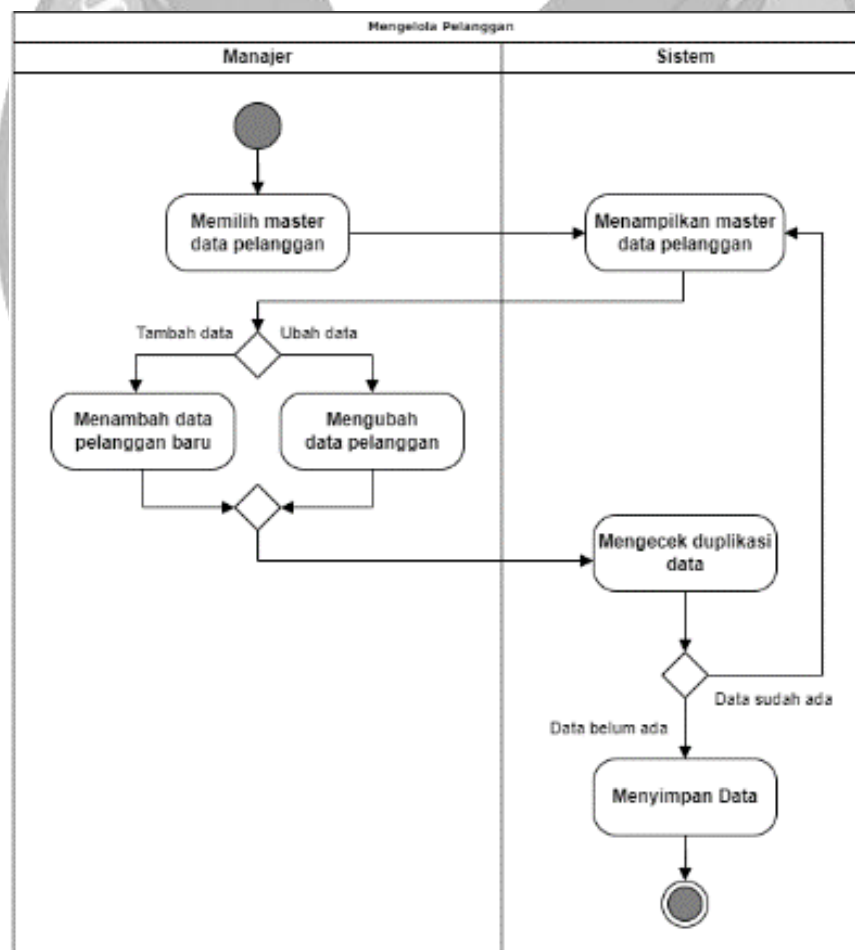


Gambar 4.14 Activity Diagram Mengelola Metode Bayar

14) Mengelola Pelanggan

Berikut adalah *activity diagram* dari Mengelola Pelanggan. Langkah-langkah proses mengelola pelanggan terdiri dari berikut:

- a) Manajer memilih halaman *master data* pelanggan.
- b) Sistem menampilkan halaman *master data* pelanggan.
- c) Jika manajer ingin menambah data baru, maka manajer memilih *form input* data baru dan menambah data pelanggan baru.
- d) Jika manajer ingin mengubah data, maka manajer memilih *form edit* data dan mengubah data pelanggan.
- e) Sistem akan mengecek duplikasi data.
- f) Jika data sudah ada di dalam *database*, maka sistem menampilkan *alert message* dan balik ke halaman sebelumnya.
- g) Jika data belum ada di dalam *database*, maka sistem akan menyimpan data pelanggan baru dan lanjut menampilkan halaman *master data* pelanggan.

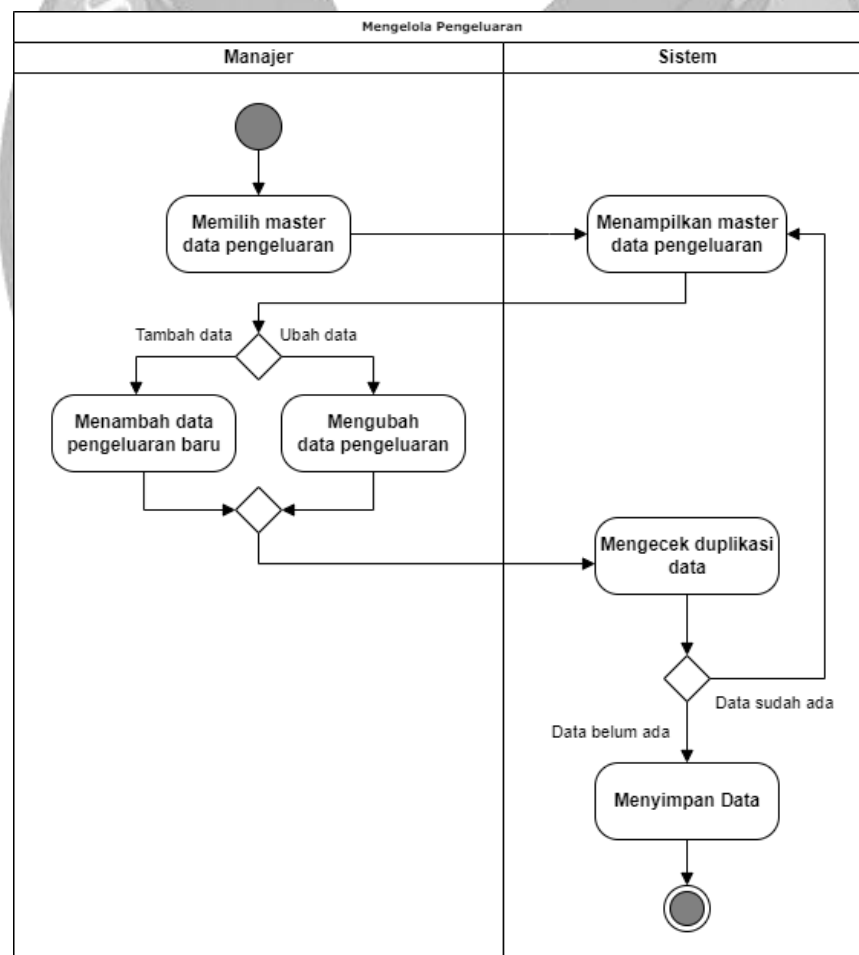


Gambar 4.15 Activity Diagram Mengelola Pelanggan

15) Mengelola Pengeluaran

Berikut adalah *activity diagram* dari Mengelola Pengeluaran. Langkah-langkah proses mengelola pengeluaran terdiri dari berikut:

- a) Manajer memilih halaman *master data* pengeluaran.
- b) Sistem menampilkan halaman *master data* pengeluaran.
- c) Jika manajer ingin menambah data baru, maka manajer memilih *form input* data baru dan menambah data pengeluaran baru.
- d) Jika manajer ingin mengubah data, maka manajer memilih *form edit* data dan mengubah data pengeluaran.
- e) Sistem akan mengecek duplikasi data.
- f) Jika data sudah ada di dalam *database*, maka sistem menampilkan *alert message* dan balik ke halaman sebelumnya.
- g) Jika data belum ada di dalam *database*, maka sistem akan menyimpan data pengeluaran baru dan lanjut menampilkan halaman *master data* pengeluaran.

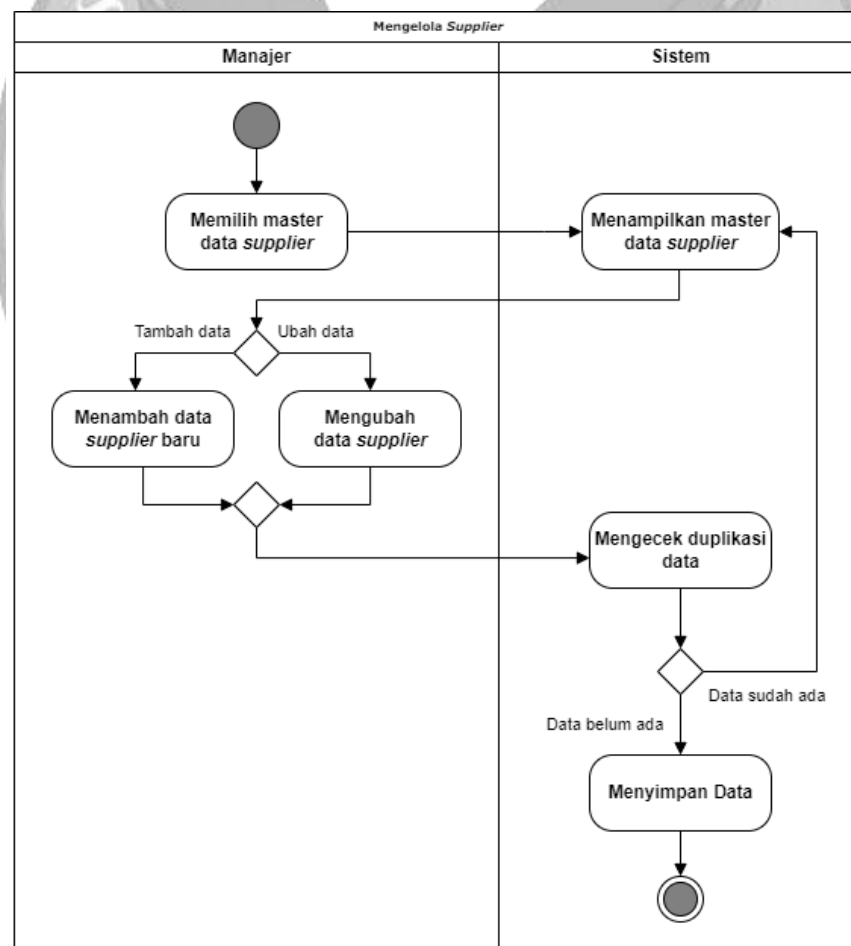


Gambar 4.16 Activity Diagram Mengelola Pengeluaran

16) Mengelola *Supplier*

Berikut adalah *activity diagram* dari Mengelola *Supplier*. Langkah-langkah proses mengelola *supplier* terdiri dari berikut:

- Manajer memilih halaman *master data supplier*.
- Sistem menampilkan halaman *master data supplier*.
- Jika manajer ingin menambah data baru, maka manajer memilih *form input* data baru dan menambah data *supplier* baru.
- Jika manajer ingin mengubah data, maka manajer memilih *form edit* data dan mengubah data *supplier*.
- Sistem akan mengecek duplikasi data.
- Jika data sudah ada di dalam *database*, maka sistem menampilkan *alert message* dan balik ke halaman sebelumnya.
- Jika data belum ada di dalam *database*, maka sistem akan menyimpan data *supplier* baru dan lanjut menampilkan halaman *master data supplier*.

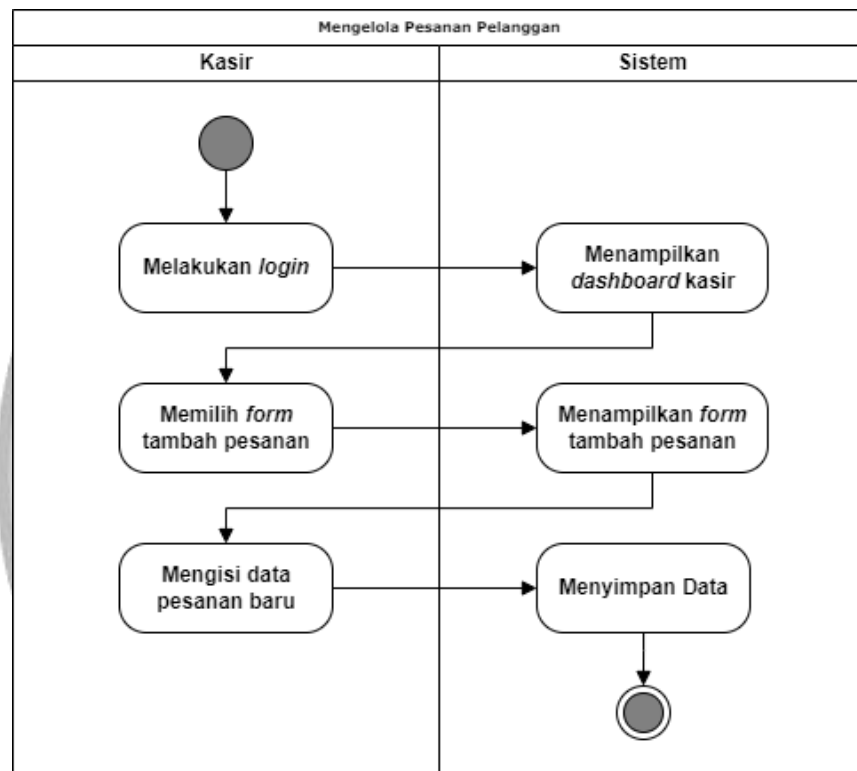


Gambar 4.17 Activity Diagrami Mengelola *Supplier*

17) Mengelola Pesanan Pelanggan

Berikut adalah *activity diagram* dari Mengelola Pesanan Pelanggan. Langkah-langkah proses mengelola pesanan pelanggan terdiri dari berikut:

- Kasir melakukan *log in*.
- Sistem menampilkan halaman *dashboard* kasir.
- Kasir memilih *form* tambah pesanan.
- Sistem menampilkan *form* tambah pesanan.
- Kasir menambah data pesanan baru.
- Sistem menyimpan data pesanan ke *database*.

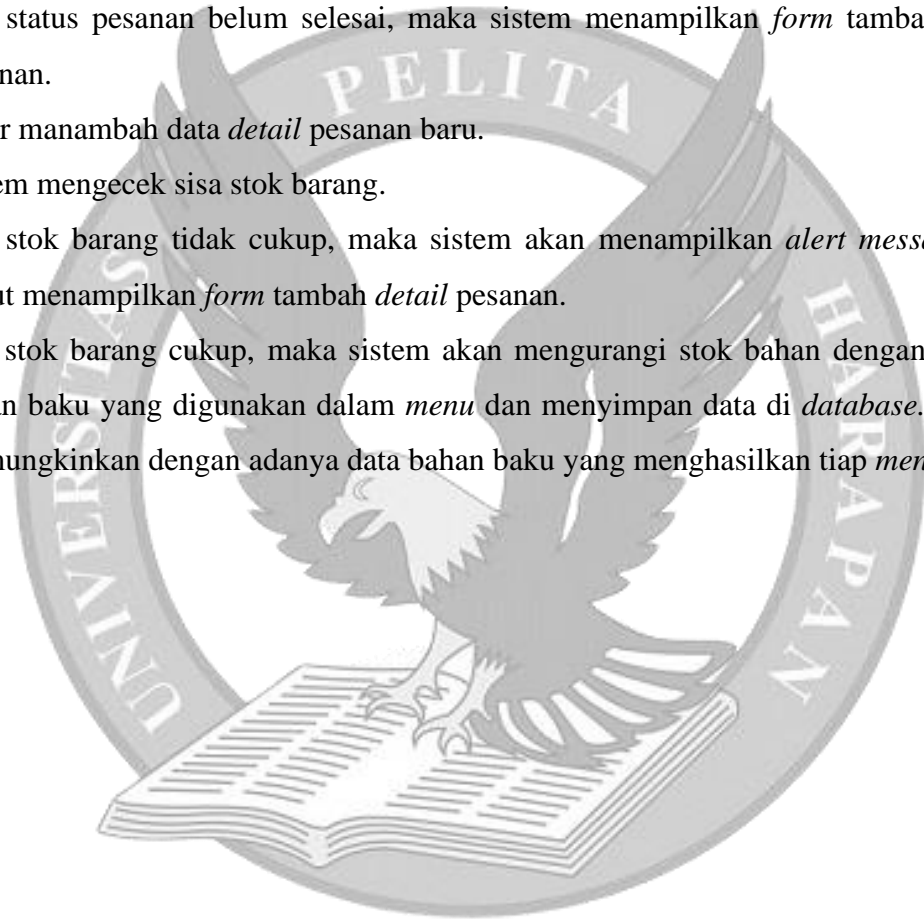


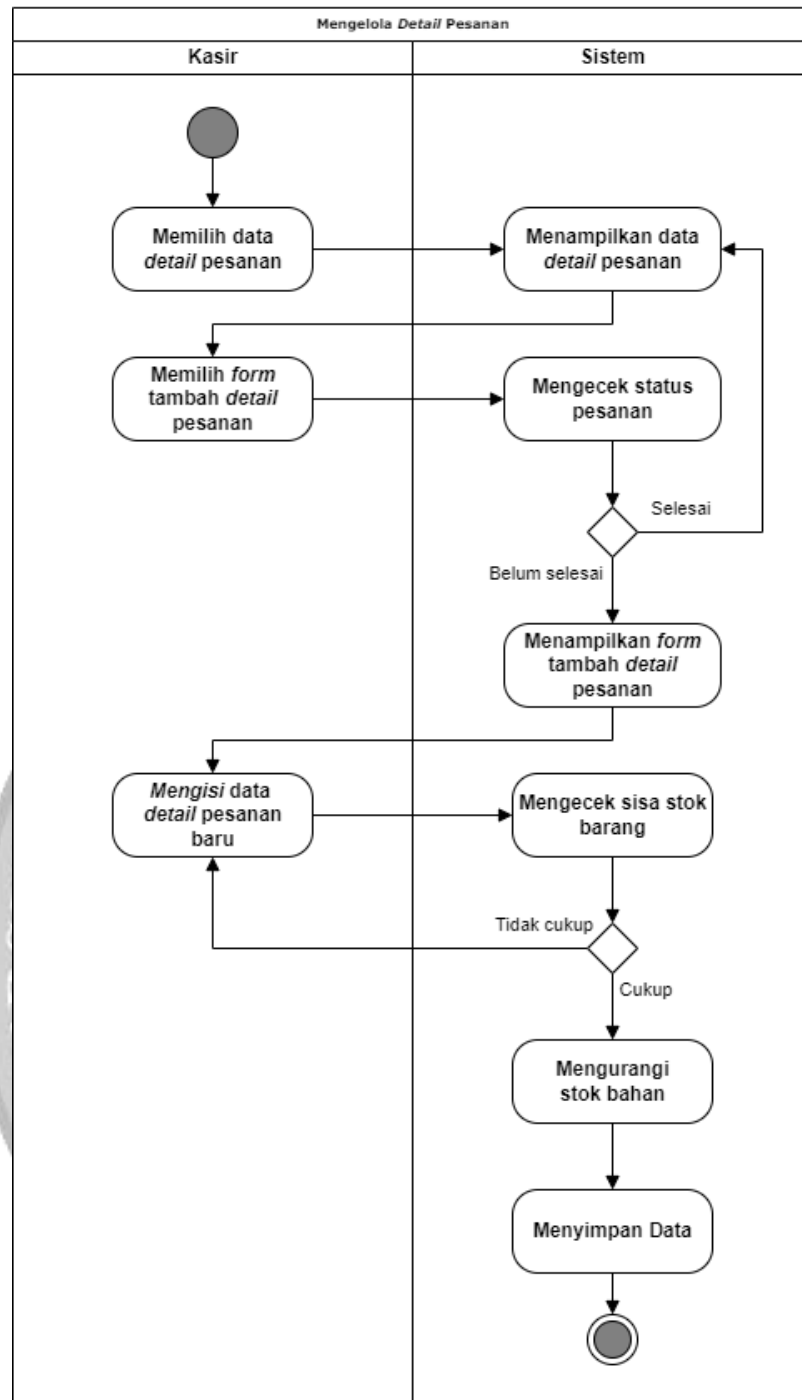
Gambar 4.18 Activity Diagram Mengelola Pesanan Pelanggan

18) Mengelola *Detail* Pesanan

Berikut adalah *activity diagram* dari Mengelola *Detail* Pesanan. Langkah-langkah proses mengelola *detail* pesanan terdiri dari berikut:

- a) Kasir memilih halaman *detail* pesanan.
- b) Sistem menampilkan halaman *detail* pesanan.
- c) Kasir memilih *form* tambah *detail* pesanan.
- d) Sistem mengecek status pesanan.
- e) Jika status pesanan sudah selesai, maka sistem menampilkan *alert message* dan lanjut menampilkan halaman *detail* pesanan.
- f) Jika status pesanan belum selesai, maka sistem menampilkan *form* tambah *detail* pesanan.
- g) Kasir menambahkan data *detail* pesanan baru.
- h) Sistem mengecek sisa stok barang.
- i) Jika stok barang tidak cukup, maka sistem akan menampilkan *alert message* dan lanjut menampilkan *form* tambah *detail* pesanan.
- j) Jika stok barang cukup, maka sistem akan mengurangi stok bahan dengan jumlah bahan baku yang digunakan dalam *menu* dan menyimpan data di *database*. Hal ini memungkinkan dengan adanya data bahan baku yang menghasilkan tiap *menu*.



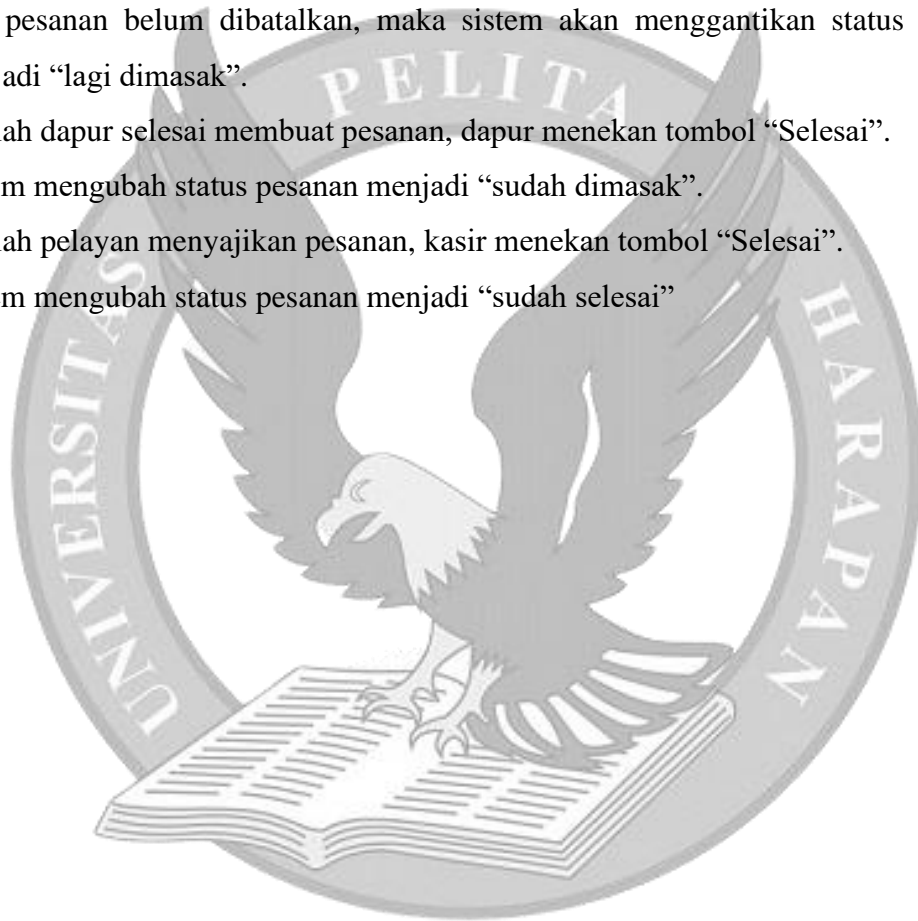


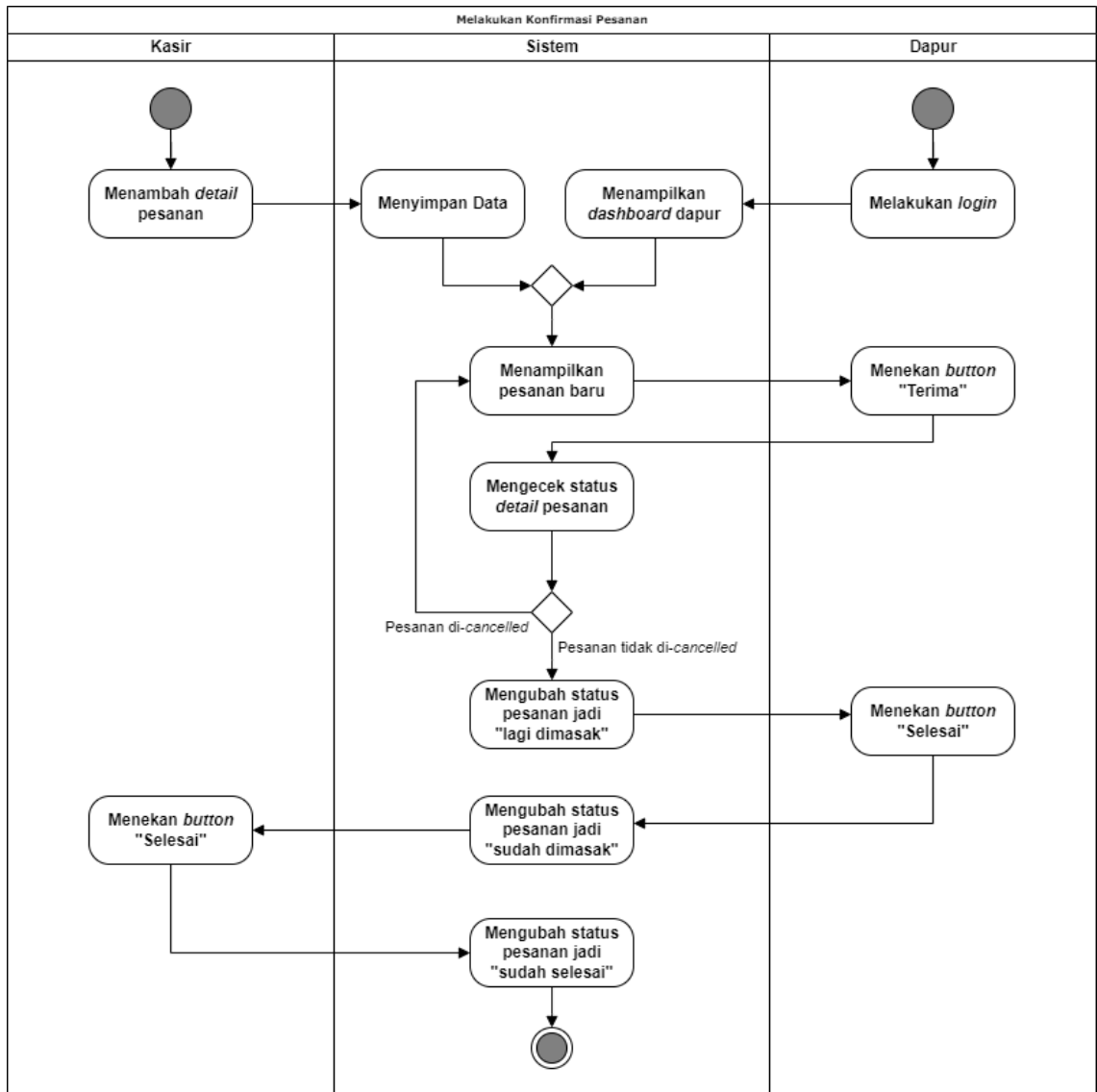
Gambar 4.19 Activity Diagram Mengelola Detail Pesanan

19) Melakukan Konfirmasi Pesanan

Berikut adalah *activity diagram* dari Melakukan Konfirmasi Pesanan. Langkah-langkah proses melakukan konfirmasi pesanan terdiri dari berikut:

- a) Dapur melakukan *log in*.
- b) Sistem menampilkan *dashboard* dapur, yang menampilkan tabel yang terdiri dari pesanan baru yang ditambah oleh pihak kasir.
- c) Dapur menekan tombol “Terima”.
- d) Sistem mengecek status pesanan.
- e) Jika pesanan sudah dibatalkan, maka sistem menampilkan *alert message*.
- f) Jika pesanan belum dibatalkan, maka sistem akan menggantikan status pesanan menjadi “lagi dimasak”.
- g) Setelah dapur selesai membuat pesanan, dapur menekan tombol “Selesai”.
- h) Sistem mengubah status pesanan menjadi “sudah dimasak”.
- i) Setelah pelayan menyajikan pesanan, kasir menekan tombol “Selesai”.
- j) Sistem mengubah status pesanan menjadi “sudah selesai”



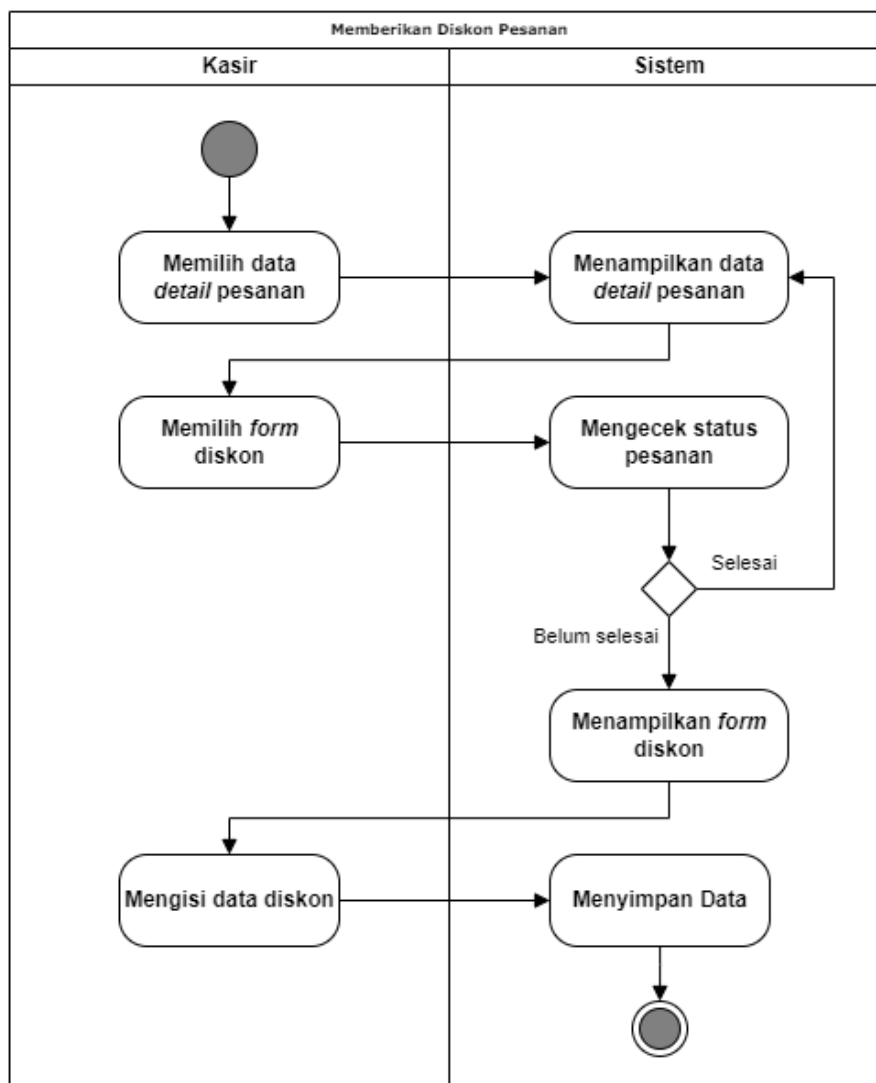


Gambar 4.20 Activity Diagram Melakukan Konfirmasi Pesanan

20) Memberikan Diskon Pesanan

Berikut adalah *activity diagram* dari Memberikan Diskon Pesanan. Langkah-langkah proses memberikan diskon pesanan terdiri dari berikut:

- a) Kasir memilih halaman *data detail* pesanan.
- b) Sistem menampilkan halaman *data detail* pesanan.
- c) Kasir memilih *form* diskon.
- d) Sistem mengecek status pesanan.
- e) Jika status pesanan sudah selesai, maka sistem menampilkan *alert message* dan lanjut menampilkan *data detail* pesanan.
- f) Jika status pesanan belum selesai, maka sistem menampilkan *form* diskon.
- g) Kasir mengisi jumlah diskon.
- h) Sistem akan menyimpan data diskon.

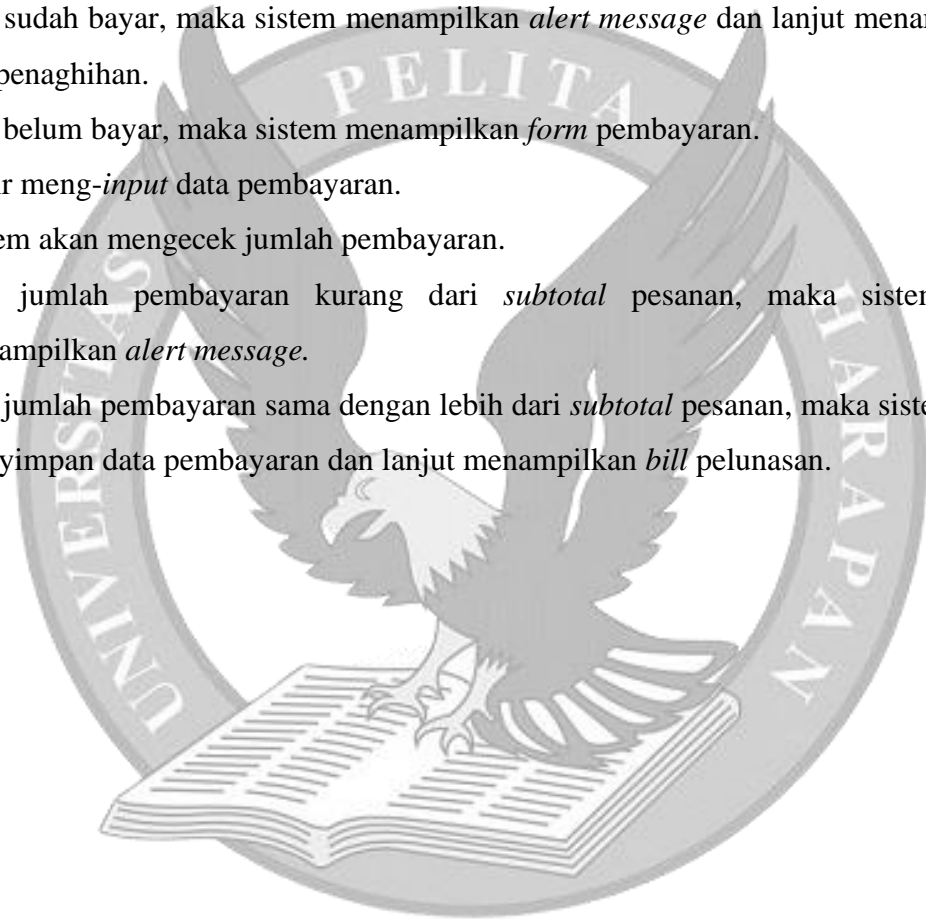


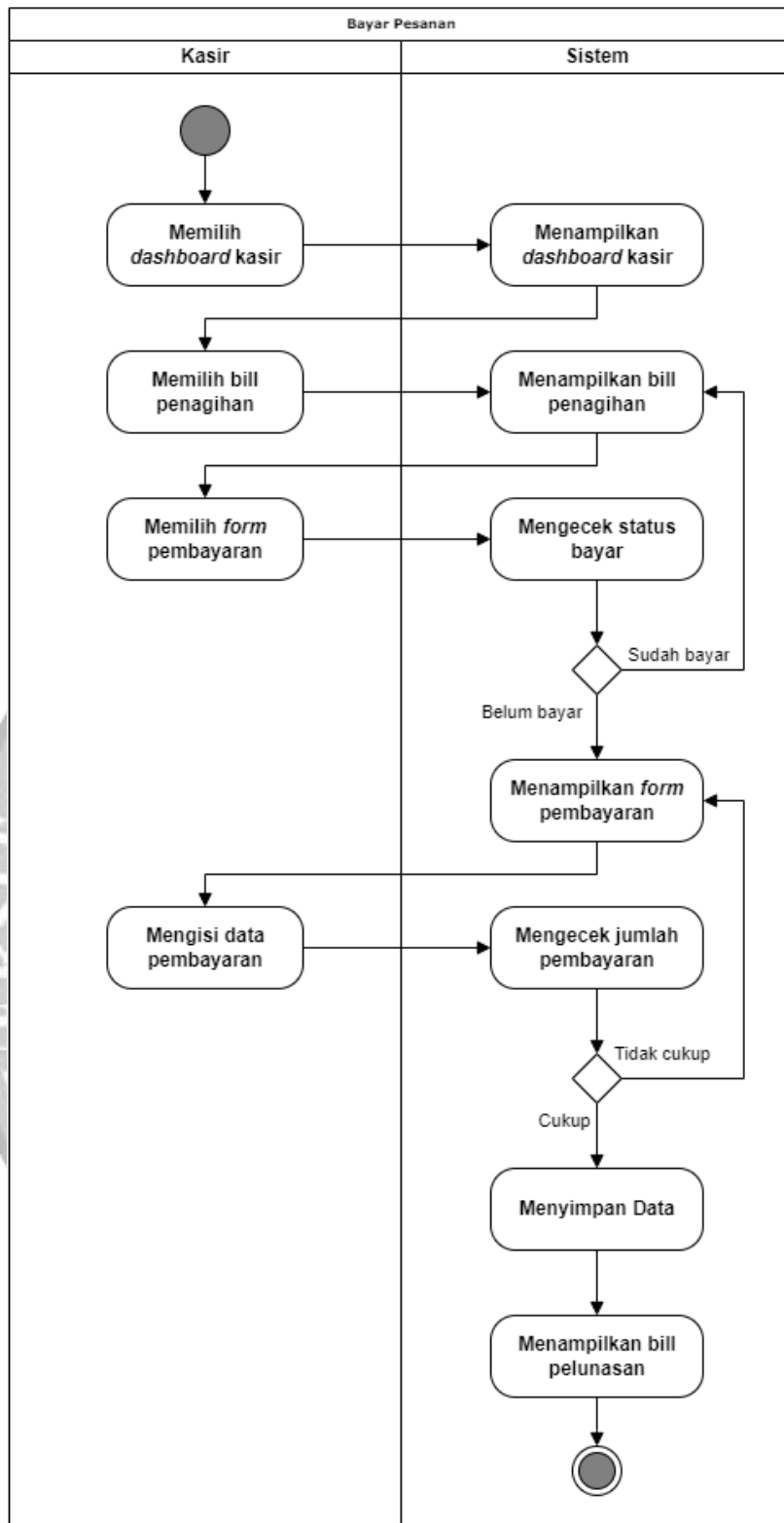
Gambar 4.21 *Activity Diagram* Memberikan Diskon Pesanan

21) Bayar Pesanan

Berikut adalah *activity diagram* dari Bayar Pesanan. Langkah-langkah proses bayar pesanan terdiri dari berikut:

- a) Kasir memilih *dashboard* kasir.
- b) Sistem menampilkan *dashboard* kasir.
- c) Kasir memilih *bill* penagihan.
- d) Sistem menampilkan *bill* penagihan.
- e) Kasir memilih *form* pembayaran.
- f) Sistem memeriksa status pembayaran.
- g) Jika sudah bayar, maka sistem menampilkan *alert message* dan lanjut menampilkan *bill* penagihan.
- h) Jika belum bayar, maka sistem menampilkan *form* pembayaran.
- i) Kasir meng-*input* data pembayaran.
- j) Sistem akan mengecek jumlah pembayaran.
- k) Jika jumlah pembayaran kurang dari *subtotal* pesanan, maka sistem akan menampilkan *alert message*.
- l) Jika jumlah pembayaran sama dengan lebih dari *subtotal* pesanan, maka sistem akan menyimpan data pembayaran dan lanjut menampilkan *bill* pelunasan.



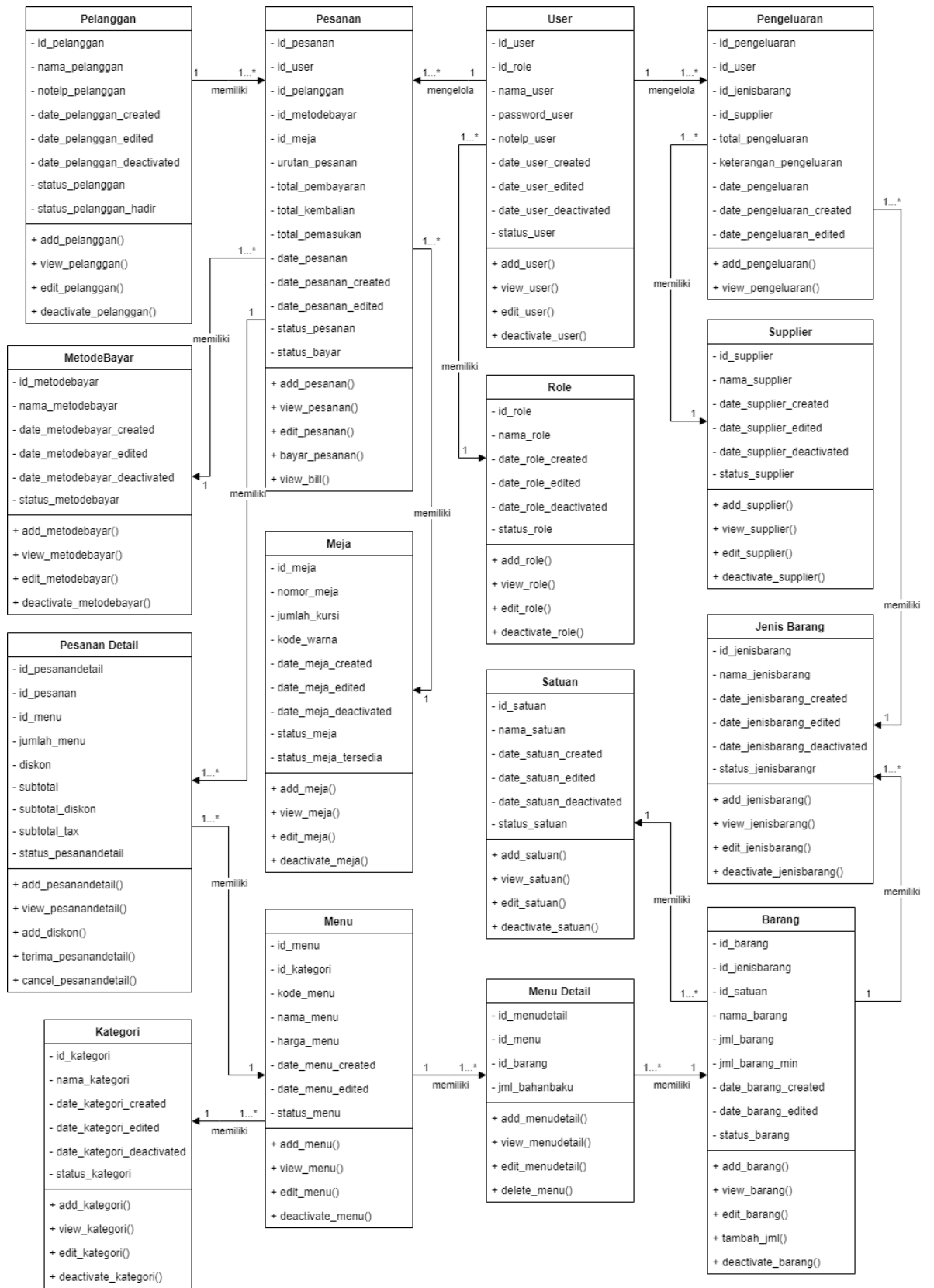


Gambar 4.22 Activity Diagram Bayar Pesanan

4.3.2. Pemodelan Struktural

Pemodelan struktural digunakan untuk memberikan gambaran struktur data pada sistem. Pemodelan struktural pada sistem yang dirancang akan ditunjukkan dalam bentuk *class diagram*.





Gambar 4.23 Class Diagram Sistem Usulan

4.3.3.2. Table Description

Berikut adalah deskripsi tabel-tabel yang digunakan dalam *database* sistem yang diusulkan:

1) Tabel *User*

Tabel 4.3 Deskripsi Tabel *User*

<i>Field Name</i>	<i>Type</i>	<i>Length</i>	<i>Key</i>
id_user	varchar	6	<i>Primary Key</i>
nama_user	text	-	
password_user	text	-	
notelp_user	text	-	
date_user_created	datetime	-	
date_user_edited	datetime	-	
date_user_deactivated	datetime	-	
status_user	text	-	
id_role	varchar	6	<i>Foreign Key</i>

2) Tabel *Role*

Tabel 4.4 Deskripsi Tabel *Role*

<i>Field Name</i>	<i>Type</i>	<i>Length</i>	<i>Key</i>
id_role	varchar	6	<i>Primary Key</i>
nama_role	text	-	
date_role_created	datetime	-	
date_role_edited	datetime	-	
date_role_deactivated	datetime	-	
status_role	text	-	

3) Tabel Barang

Tabel 4.5 Deskripsi Tabel Barang

<i>Field Name</i>	<i>Type</i>	<i>Length</i>	<i>Key</i>
id_barang	varchar	6	<i>Primary Key</i>
nama_barang	text	-	
jml_barang	float	-	
jml_barang_min	float	-	
date_barang_created	datetime	-	
date_barang_edited	datetime	-	
status_barang	text	-	
id_jenisbarang	varchar	6	<i>Foreign Key</i>
id_satuan	varchar	6	<i>Foreign Key</i>

4) Tabel Jenis Barang

Tabel 4.6 Deskripsi Tabel Jenis Barang

<i>Field Name</i>	<i>Type</i>	<i>Length</i>	<i>Key</i>
id_jenisbarang	varchar	6	<i>Primary Key</i>
nama_jenisbarang	text	-	
date_jenisbarang_created	datetime	-	
date_jenisbarang_edited	datetime	-	
date_jenisbarang_deactivated	datetime	-	
status_jenisbarang	text	-	

5) Tabel Satuan

Tabel 4.7 Deskripsi Tabel Satuan

<i>Field Name</i>	<i>Type</i>	<i>Length</i>	<i>Key</i>
id_satuan	varchar	6	<i>Primary Key</i>
nama_satuan	text	-	
date_satuan_created	datetime	-	
date_satuan_edited	datetime	-	
date_satuan_deactivated	datetime	-	
status_satuan	text	-	

6) Tabel Menu

Tabel 4.8 Deskripsi Tabel Menu

<i>Field Name</i>	<i>Type</i>	<i>Length</i>	<i>Key</i>
id_menu	varchar	6	<i>Primary Key</i>
kode_menu	int	-	
nama_menu	text	-	
harga_menu	int	-	
date_menu_created	datetime	-	
date_menu_edited	datetime	-	
status_menu	text	-	
id_kategori	varchar	6	<i>Foreign Key</i>

7) Tabel Menu Detail

Tabel 4.9 Deskripsi Tabel Menu Detail

<i>Field Name</i>	<i>Type</i>	<i>Length</i>	<i>Key</i>
id_menudetail	varchar	20	<i>Primary Key</i>
jml_bahanbaku	float	-	
id_menu	varchar	6	<i>Foreign Key</i>
id_barang	varchar	6	<i>Foreign Key</i>

8) Tabel Kategori

Tabel 4.10 Deskripsi Tabel Kategori

<i>Field Name</i>	<i>Type</i>	<i>Length</i>	<i>Key</i>
id_kategori	varchar	6	<i>Primary Key</i>
nama_kategori	text	-	
date_kategori_created	datetime	-	
date_kategori_edited	datetime	-	
date_kategori_deactivated	datetime	-	
status_kategori	text	-	

9) Tabel Pengeluaran

Tabel 4.11 Deskripsi Tabel Pengeluaran

<i>Field Name</i>	<i>Type</i>	<i>Length</i>	<i>Key</i>
id_pengeluaran	varchar	20	<i>Primary Key</i>
total_pengeluaran	Int	-	
keterangan_pengeluaran	text	-	
date_pengeluaran	date	-	
date_pengeluaran_created	datetime	-	
date_pengeluaran_edited	datetime	-	
id_user	varchar	6	<i>Foreign Key</i>
id_jenisbarang	varchar	6	<i>Foreign Key</i>
id_supplier	varchar	6	<i>Foreign Key</i>

10) Tabel Supplier

Tabel 4.12 Deskripsi Tabel Supplier

<i>Field Name</i>	<i>Type</i>	<i>Length</i>	<i>Key</i>
id_supplier	varchar	6	<i>Primary Key</i>
nama_supplier	text	-	
date_supplier_created	datetime	-	
date_supplier_edited	datetime	-	
date_supplier_deactivated	datetime	-	
status_supplier	text	-	

11) Tabel Pesanan

Tabel 4.13 Deskripsi Tabel Pesanan

<i>Field Name</i>	<i>Type</i>	<i>Length</i>	<i>Key</i>
id_pesanan	varchar	20	<i>Primary Key</i>
urutan_pesanan	int	-	
total_pembayaran	int	-	
total_kembalian	int	-	
total_pemasukan	int	-	
date_pesanan	date	-	
date_pesanan_created	datetime	-	
date_pesanan_edited	datetime	-	
status_pesanan	text	-	
status_bayar	text	-	
id_user	varchar	6	<i>Foreign Key</i>
id_pelanggan	varchar	10	<i>Foreign Key</i>
id_metodebayar	varchar	6	<i>Foreign Key</i>
id_meja	varchar	6	<i>Foreign Key</i>

12) Tabel Pesanan *Detail*

Tabel 4.14 Deskripsi Tabel Pesanan *Detail*

<i>Field Name</i>	<i>Type</i>	<i>Length</i>	<i>Key</i>
id_pesanandetail	varchar	20	<i>Primary Key</i>
jumlah_menu	int	-	
diskon	tinyint	-	
subtotal	int	-	
subtotal_diskon	int	-	
subtotal_tax	int	-	
status_pesanandetail	Text	-	
id_pesanan	varchar	20	<i>Foreign Key</i>
id_menu	varchar	6	<i>Foreign Key</i>

13) Tabel Meja

Tabel 4.15 Deskripsi Tabel Meja

<i>Field Name</i>	<i>Type</i>	<i>Length</i>	<i>Key</i>
id_meja	varchar	6	<i>Primary Key</i>
nomor_meja	text	-	
jumlah_kursi	int	-	
date_meja_created	datetime	-	
date_meja_edited	datetime	-	
date_meja_deactivated	datetime	-	
status_meja	text	-	
status_meja_tersedia	text	-	

14) Tabel Metode Bayar

Tabel 4.16 Deskripsi Tabel Metode Bayar

<i>Field Name</i>	<i>Type</i>	<i>Length</i>	<i>Key</i>
id_metodebayar	varchar	6	<i>Primary Key</i>
nama_metodebayar	text	-	
date_metodebayar_created	datetime	-	
date_metodebayar_edited	datetime	-	
date_metodebayar_deactivated	datetime	-	
status_metodebayar	text	-	

15) Tabel Pelanggan

Tabel 4.17 Deskripsi Tabel Pelanggan

<i>Field Name</i>	<i>Type</i>	<i>Length</i>	<i>Key</i>
id_pelanggan	varchar	6	<i>Primary Key</i>
nama_pelanggan	text	-	
notelp_pelanggan	text	-	
date_pelanggan_created	datetime	-	
date_pelanggan_edited	datetime	-	
date_pelanggan_deactivated	datetime	-	
status_pelanggan	text	-	

4.4. Implementasi Sistem

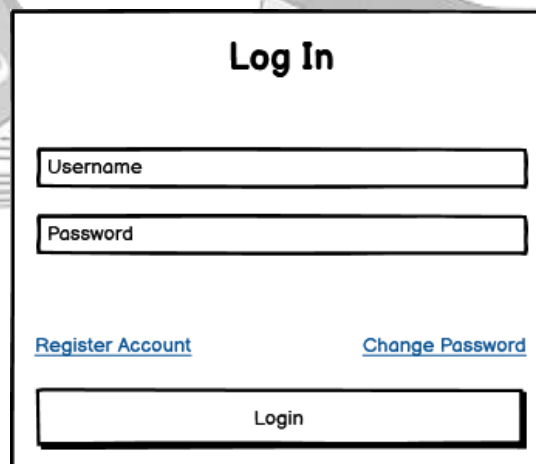
Implementasi sistem pencatatan keuangan dan inventori barang usulan terdiri dari tahap perancangan *user interface* dan tahap pengujian.

4.4.1. Mockup

Berikut adalah kumpulan dari rancangan *mockup* sistem usulan.

1) Login

Gambar berikut adalah rancangan *user interface* pada halaman *login*.



Log In

Username

Password

[Register Account](#) [Change Password](#)

Login

Gambar 4.25 Rancangan Halaman *Log In*

2) Register

Gambar berikut adalah rancangan *user interface* pada halaman *register*.

Sign Up

Username

Nomor Telepon

Password

Role

Kasir

Dapur

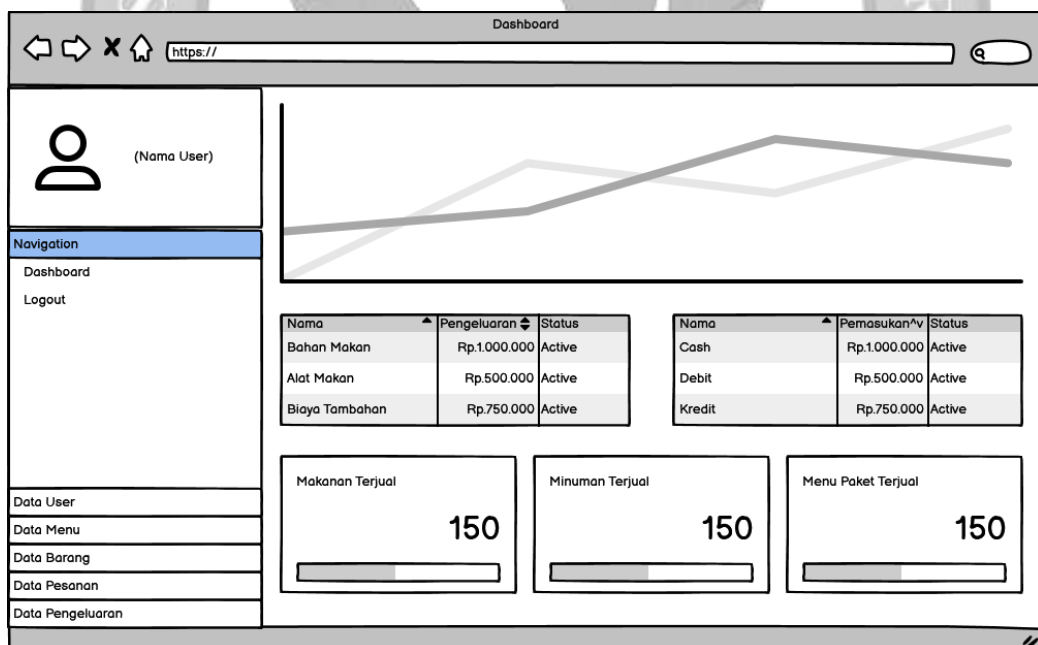
Register

[Login](#)

Gambar 4.26 Rancangan Halaman *Sign Up*

3) Dashboard (Manajer)

Gambar berikut adalah rancangan *user interface* pada halaman *dashboard* manajer.



Gambar 4.27 Rancangan Halaman *Dashboard* (Manajer)

4) *Dashboard* (Kasir)

Gambar berikut adalah rancangan *user interface* pada halaman *dashboard* kasir.

Tanggal Hari Ini : 01-01-2022

Tambah Pesanan

Pesanan #1

01/01/2022 - 12.00.00

Status : sudah dianter

Tamu : Guest

1

Detail Tambah Bayar Bill

Gambar 4.28 Rancangan Halaman *Dashboard* (Kasir)

5) *Dashboard* (Dapur)

Gambar berikut adalah rancangan *user interface* pada halaman *dashboard* dapur.

Tanggal Hari Ini : 01-01-2022

Pesanan	Meja	Menu	Jumlah	Status	Aksi
1	1	teh manis	1	belum diterima	Terima Selesai

Gambar 4.29 Rancangan Halaman *Dashboard* (Dapur)

6) Mengelola *User*

Gambar berikut adalah rancangan *user interface* pada halaman *master data user*.

Halaman hanya dapat dibuka oleh pihak manajer.

Data User

Tambah PDF

Q search

Nama^v	No. Telp^v	Role^v	Aksi
John Doe	081234567890	Manajer	<input type="button" value="Edit"/> <input type="button" value="Deactivate"/>

Gambar 4.30 Rancangan Halaman *Master Data User*

Gambar berikut adalah rancangan *user interface* pada halaman untuk menambah data *user*.

Input Data User

Nama

Password

Nomor Telepon

Role

Gambar 4.31 Rancangan Halaman Menambah Data *User*

Gambar berikut adalah rancangan *user interface* pada halaman untuk mengubah data *user*.

Edit Data User

Nama

Nomor Telepon

Role

Gambar 4.32 Rancangan Halaman Mengubah Data *User*

7) Mengelola *Role*

Gambar berikut adalah rancangan *user interface* pada halaman *master data role*. Halaman hanya dapat dibuka oleh pihak manajer.

Data Role

Nama^v	Aksi
manajer	<input type="button" value="Edit"/> <input type="button" value="Deactivate"/>

Gambar 4.33 Rancangan Halaman *Master Data Role*

Gambar berikut adalah rancangan *user interface* pada halaman untuk menambah data *role*.

Gambar 4.34 Rancangan Halaman Menambah Data *Role*

Gambar berikut adalah rancangan *user interface* pada halaman untuk mengubah data *role*.

Gambar 4.35 Rancangan Halaman Mengubah Data *Role*

8) Mengelola *Menu*

Gambar berikut adalah rancangan *user interface* pada halaman *master data menu*. Halaman hanya dapat diakses oleh pihak manajer.

Kode ^v	Nama ^v	Harga ^v	Kategori ^v	Status ^v	Aksi
101	teh manis	Rp.10.000	minuman	active	Detail Edit Deactivate

Gambar 4.36 Perancangan Halaman *Master Data Menu*

Gambar berikut adalah rancangan *user interface* pada halaman untuk menambah data *menu*.

Input Data Menu	
Kode	<input type="text" value="Kode Menu"/>
Nama	<input type="text" value="Nama Menu"/>
Harga	<input type="text" value="Harga Menu"/>
Kategori	<input type="text" value="Kategori"/>
<input type="button" value="Submit"/> <input type="button" value="Back"/>	

Gambar 4.37 Perancangan Halaman Menambah Data *Menu*

Gambar berikut adalah rancangan *user interface* pada halaman untuk mengubah data *menu*.

Edit Data Menu	
Kode	<input type="text" value="101"/>
Nama	<input type="text" value="teh manis"/>
Harga	<input type="text" value="10000"/>
Kategori	<input type="text" value="minuman"/>
<input type="button" value="Submit"/> <input type="button" value="Back"/>	

Gambar 4.38 Perancangan Halaman Mengubah Data *Menu*

9) Mengelola *Detail Menu*

Gambar berikut adalah rancangan *user interface* pada halaman data *detail menu*. Halaman hanya dapat diakses oleh pihak manajer.

Menu : (nama menu)

Tambah PDF

search

Bahan ^v	Jumlah Bahan ^v	Satuan ^v	Aksi
air	500.00	milliliter	Edit Deactivate

Gambar 4.39 Perancangan Halaman Data *Detail Menu*

Gambar berikut adalah rancangan *user interface* pada halaman untuk menambah data *detail menu*.

Input Data Menu Detail

Nama
(nama menu)

Bahan
bahan

Jumlah
Jumlah Barang

Submit Back

Gambar 4.40 Perancangan Halaman Menambah Data *Detail Menu*

Gambar berikut adalah rancangan *user interface* pada halaman untuk mengubah *detail menu*.

Edit Data Menu Detail

Menu

Bahan

Jumlah

Gambar 4.41 Perancangan Halaman Mengubah Data *Detail Menu*

10) Mengelola Kategori

Gambar berikut adalah rancangan *user interface* pada halaman *master data* kategori. Halaman hanya dapat diakses oleh pihak manajer.

Data Kategori

Nama	Aksi
minuman	<input type="button" value="Edit"/> <input type="button" value="Deactivate"/>

Gambar 4.42 Perancangan Halaman *Master Data* Kategori

Gambar berikut adalah rancangan *user interface* pada halaman untuk menambah data kategori.

Gambar 4.43 Perancangan Halaman Menambah Data Kategori

Gambar berikut adalah rancangan *user interface* pada halaman untuk mengubah data kategori.

Gambar 4.44 Perancangan Halaman Mengubah Data Kategori

11) Mengelola Barang

Gambar berikut adalah rancangan *user interface* pada halaman *master data* barang. Halaman hanya dapat diakses oleh pihak manajer.

Nama ^v	Jumlah ^v	Jumlah Minimal ^v	Satuan ^v	Jenis Barang ^v	Status ^v	Aksi
air	1000.00	600.00	mililiter	bahan makan	active	Add Edit Deactivate

Gambar 4.45 Perancangan Halaman *Master Data* Barang

Gambar berikut adalah rancangan *user interface* pada halaman untuk menambah data barang.

Input Data Barang	
Nama	<input type="text" value="Nama Barang"/>
Jumlah	<input type="text" value="Jumlah Barang"/>
Jumlah Min.	<input type="text" value="Jumlah Barang Minimal"/>
Satuan	<input type="text" value="satuan"/>
Jenis Barang	<input type="text" value="jenis makan"/>
<input type="button" value="Submit"/> <input type="button" value="Back"/>	

Gambar 4.46 Perancangan Halaman Menambah Data Barang

Gambar berikut adalah rancangan *user interface* pada halaman untuk mengubah data barang.

Edit Data Barang	
Nama	<input type="text" value="air"/>
Jumlah	<input type="text" value="1000.00"/>
Jumlah Min.	<input type="text" value="600.00"/>
Satuan	<input type="text" value="mililiter"/>
Jenis Barang	<input type="text" value="bahan makan"/>
<input type="button" value="Submit"/> <input type="button" value="Back"/>	

Gambar 4.47 Perancangan Halaman Mengubah Data Barang

Gambar berikut adalah rancangan *user interface* pada halaman untuk menambah jumlah stok barang.

Add Data Barang

Nama
air

Jumlah
1000.00

Tambah Jumlah
0

Satuan
mililiter

Submit Back

Gambar 4.48 Perancangan Halaman Menambah Jumlah Stok Barang

12) Mengelola Jenis Barang

Gambar berikut adalah rancangan *user interface* pada halaman *master data* jenis barang. Halaman hanya dapat diakses oleh pihak manajer.

Data Jenis Barang

Tambah PDF

Q search

Nama	Aksi
bahan makanan	Edit Deactivate

Gambar 4.49 Perancangan Halaman *Master Data* Jenis Barang

Gambar berikut adalah rancangan *user interface* pada halaman untuk menambah data jenis barang.

Gambar 4.50 Perancangan Halaman Menambah Data Jenis Barang

Gambar berikut adalah rancangan *user interface* pada halaman untuk mengubah data jenis barang.

Gambar 4.51 Perancangan Halaman Mengubah Data Jenis Barang

13) Mengelola Satuan

Gambar berikut adalah rancangan *user interface* pada halaman *master data* satuan. Halaman hanya dapat diakses oleh pihak manajer.

Nama	Aksi
milliter	<input type="button" value="Edit"/> <input type="button" value="Deactivate"/>

Gambar 4. 52 Perancangan Halaman *Master Data* Satuan

Gambar berikut adalah rancangan *user interface* pada halaman untuk menambah data satuan.

Input Data Satuan

Nama

Nama Satuan

Submit Back

Gambar 4.53 Perancangan Halaman Menambah Data Satuan

Gambar berikut adalah rancangan *user interface* pada halaman untuk mengubah data satuan.

Edit Data Satuan

Nama

mililiter

Submit Back

Gambar 4.54 Perancangan Halaman Mengubah Data Satuan

14) Mengelola Meja

Gambar berikut adalah rancangan *user interface* pada halaman *master data* meja. Halaman hanya dapat diakses oleh pihak manajer.

Data Meja

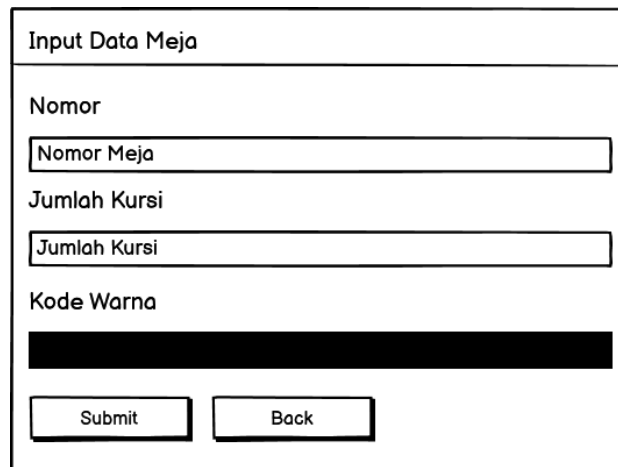
Tambah PDF

search

Nomor Meja ^v	Jumlah Kursi ^v	Kode Warna ^v	Status Tamu	Aksi
1	4	#000000	kosong	Edit Deactivate

Gambar 4.55 Perancangan Halaman *Master Data* Meja

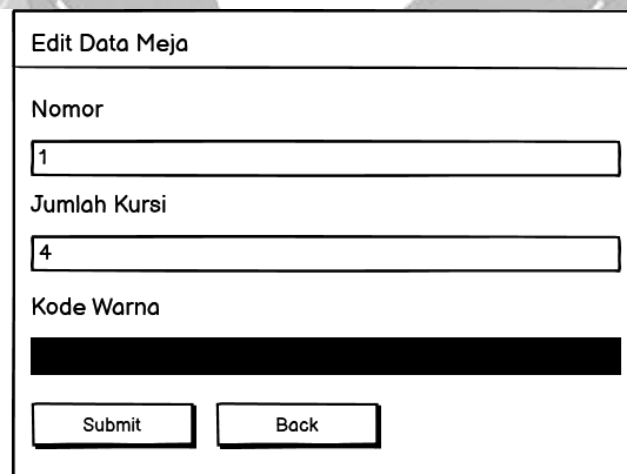
Gambar berikut adalah rancangan *user interface* pada halaman untuk menambah data meja.



The screenshot shows a web form titled "Input Data Meja". It contains three input fields: "Nomor" with the placeholder text "Nomor Meja", "Jumlah Kursi" with the placeholder text "Jumlah Kursi", and "Kode Warna" with a black color selection bar. At the bottom of the form are two buttons: "Submit" and "Back".

Gambar 4.56 Perancangan Halaman Menambah Data Meja

Gambar berikut adalah rancangan *user interface* pada halaman untuk mengubah data meja.



The screenshot shows a web form titled "Edit Data Meja". It contains three input fields: "Nomor" with the value "1", "Jumlah Kursi" with the value "4", and "Kode Warna" with a black color selection bar. At the bottom of the form are two buttons: "Submit" and "Back".

Gambar 4.57 Perancangan Halaman Mengubah Data Meja

15) Mengelola Metode Bayar

Gambar berikut adalah rancangan *user interface* pada halaman *master data* metode bayar. Halaman hanya dapat diakses oleh pihak manajer.

Data Metode Bayar

Tambah PDF

search

Nama^v	Aksi
kas	<input type="button" value="Edit"/> <input type="button" value="Deactivate"/>

Gambar 4.58 Perancangan Halaman *Master Data* Metode Bayar

Gambar berikut adalah rancangan *user interface* pada halaman untuk menambah data metode bayar.

Input Data Metode Bayar

Nama

Gambar 4.59 Perancangan Halaman Menambah Data Metode Bayar

Gambar berikut adalah rancangan *user interface* pada halaman untuk mengubah data metode bayar.

Edit Data Metode Bayar

Nama

Gambar 4.60 Perancangan Halaman Mengubah Data Metode Bayar

16) Mengelola Pelanggan

Gambar berikut adalah rancangan *user interface* pada halaman *master data* pelanggan. Halaman hanya dapat diakses oleh pihak manajer.

Nama^v	No telp^v	Status Hadir	Aksi
guest	080000000000	belum hadir	<input type="button" value="Edit"/> <input type="button" value="Deactivate"/>

Gambar 4.61 Perancangan Halaman *Master Data* Pelanggan

Gambar berikut adalah rancangan *user interface* pada halaman untuk menambah data pelanggan.

Input Data Pelanggan	
Nama	<input type="text" value="Nama Pelanggan"/>
No Telp	<input type="text" value="Nomor Telepon"/>
<input type="button" value="Submit"/>	<input type="button" value="Back"/>

Gambar 4.62 Perancangan Halaman Menambah Data Pelanggan

Gambar berikut adalah rancangan *user interface* pada halaman untuk mengubah data pelanggan.

Edit Data Pelanggan

Nama

No Telp

Gambar 4.63 Perancangan Halaman Mengubah Data Pelanggan

17) Mengelola Pengeluaran

Gambar berikut adalah rancangan *user interface* pada halaman *master data* pengeluaran. Halaman hanya dapat diakses oleh pihak manajer.

Data Pengeluaran

Tanggal Awal

Tanggal Akhir

Nama User^v	Date^v	Total Pengeluaran^v	Keterangan^v	Kategori^v	Supplier^v	Aksi
John Doe	01/01/2022	Rp.50.000	bumbu	bahan makanan	pasar pagi	<input type="button" value="Edit"/>

Gambar 4.64 Perancangan Halaman *Master Data* Pengeluaran

Gambar berikut adalah rancangan *user interface* pada halaman untuk menambah data pengeluaran.

Input Data Pengeluaran

Harga

Keterangan

Kategori

Supplier

Tanggal

Gambar 4.65 Perancangan Halaman Menambah Data Pengeluaran

Gambar berikut adalah rancangan *user interface* pada halaman untuk mengubah data pengeluaran.

Edit Data Pengeluaran

Harga

Keterangan

Kategori

Supplier

Tanggal

Gambar 4.66 Perancangan Halaman Mengubah Data Pengeluaran

18) Mengelola *Supplier*

Gambar berikut adalah rancangan *user interface* pada halaman *master data supplier*. Halaman hanya dapat diakses oleh pihak manajer.

Nama^v	Aksi
pasar pagi	<input type="button" value="Edit"/> <input type="button" value="Deactivate"/>

Gambar 4.67 Perancangan Halaman *Master Data Supplier*

Gambar berikut adalah rancangan *user interface* pada halaman untuk menambah *data supplier*.

Input Data Supplier
Nama
<input type="text" value="Nama Supplier"/>
<input type="button" value="Submit"/> <input type="button" value="Back"/>

Gambar 4.68 Perancangan Halaman Menambah *Data Supplier*

Gambar berikut adalah rancangan *user interface* pada halaman untuk mengubah *data supplier*.

Edit Data Supplier

Nama

pasar pagi

Submit

Back

Gambar 4.69 Perancangan Halaman Mengubah Data *Supplier*

19) Mengelola Pesanan

Gambar berikut adalah rancangan *user interface* pada halaman *master data* pesanan. Halaman hanya dapat diakses oleh pihak manajer.

Data Pesanan

PDF

Sort

Tanggal Awal

/ /

Tanggal Akhir

/ /

Tanggal^v	Pembayaran^v	Kembalian^v	Pemasukan^v	Urutan^v	Meja^v	Pelanggan^v	Metode Bayar^v	Kasir^v	Status^v	Aksi
01/01/2022 - 12:00:00	Rp.50.000	Rp.0	Rp.50.000	1	1	guest	kas	John Doe	sudah bayar	<div style="border: 1px solid black; padding: 2px 5px;">Detail</div>

Gambar 4.70 Perancangan Halaman *Master Data* Pesanan

Gambar berikut adalah rancangan *user interface* pada halaman untuk menambah data pesanan. Halaman dapat diakses oleh pihak kasir.

Input Data Pesanan

Meja

1

Pelanggan

guest

Submit

Back

Gambar 4.71 Perancangan Halaman Menambah Data Pesanan

20) Mengelola Pesanan *Detail*

Gambar berikut adalah rancangan *user interface* pada halaman data *detail* pesanan pihak manajer. Halaman hanya dapat diakses oleh pihak manajer.

Pesanan #1

Meja 1
01/01/2022 - 12:00:00

Menu ^v	Jumlah ^v	Subtotal ^v	Diskon(% ^v) ^v	Subtotal (Diskon) ^v	Subtotal (Tax) ^v
teh manis	1	Rp.10.000	0	Rp.10.000	Rp.11.000

Gambar 4.72 Perancangan Halaman Data *Detail* Pesanan (Manajer)

Gambar berikut adalah rancangan *user interface* pada halaman data *detail* pesanan pihak kasir. Halaman dapat diakses oleh pihak kasir.

Pesanan #1

Meja 1
01/01/2022 - 12:00:00

Menu ^v	Jumlah ^v	Subtotal ^v	Diskon(% ^v) ^v	Subtotal (Diskon) ^v	Subtotal (Tax) ^v	Status ^v	Aksi
teh manis	1	Rp.10.000	0	Rp.10.000	Rp.11.000	belum diterima	<input type="button" value="Diskon"/> <input type="button" value="Selesai"/> <input type="button" value="Cancel"/>

Gambar 4.73 Perancangan Halaman Data *Detail* Pesanan (Kasir)

Gambar berikut adalah rancangan *user interface* pada halaman untuk menambah data *detail* pesanan.

Input Data Pesanan #1

Menu

teh manis

Jumlah

1

Submit Back

Gambar 4.74 Perancangan Halaman Menambah Data Pesanan *Detail*

Gambar berikut adalah rancangan *user interface* pada halaman untuk memberikan diskon kepada pesanan.

Add Diskon

Nama Menu

teh manis

Diskon (%)

0

Submit Back

Gambar 4.75 Perancangan Halaman Menambah Diskon Pesanan

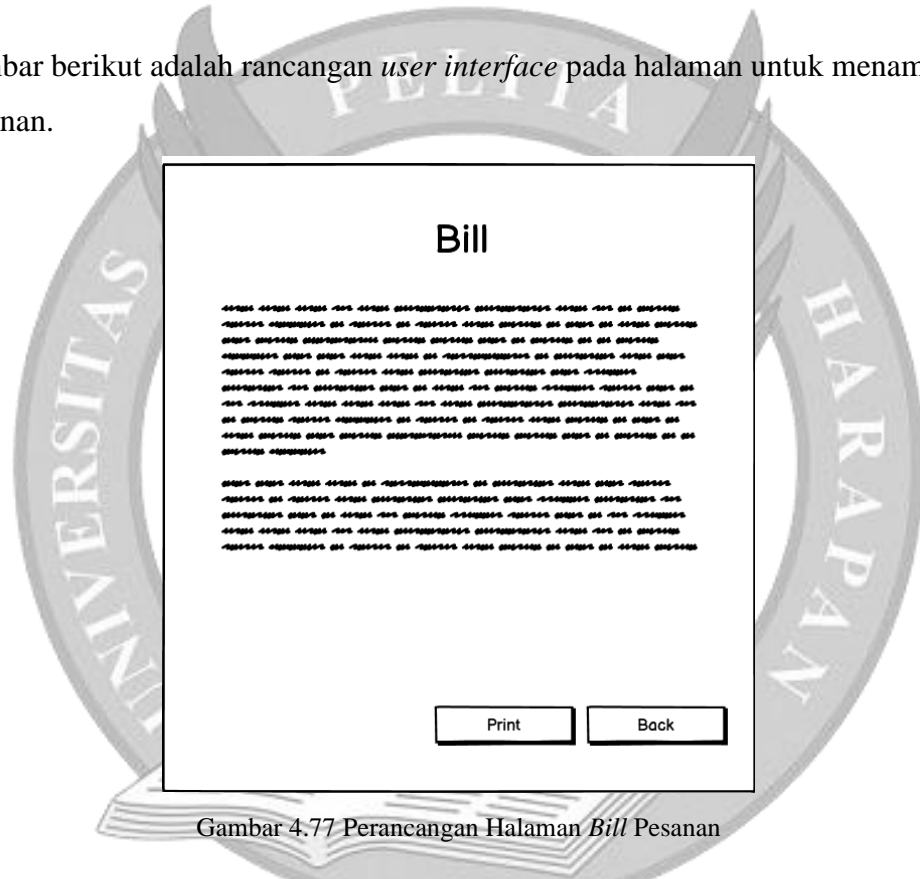
21) Pembayaran Pesanan

Gambar berikut adalah rancangan *user interface* pada halaman untuk melakukan pembayaran.

Subtotal : Rp.10.000,-	
Bayar	
<input type="text" value="Pembayaran"/>	
Metode Bayar	
<input type="text" value="kas"/>	
<input type="button" value="Submit"/>	<input type="button" value="Back"/>

Gambar 4.76 Perancangan Halaman Bayar Pesanan

Gambar berikut adalah rancangan *user interface* pada halaman untuk menampilkan bill pesanan.



Gambar 4.77 Perancangan Halaman *Bill* Pesanan

4.4.2. Perancangan Lapisan Antarmuka (*User Interface*)

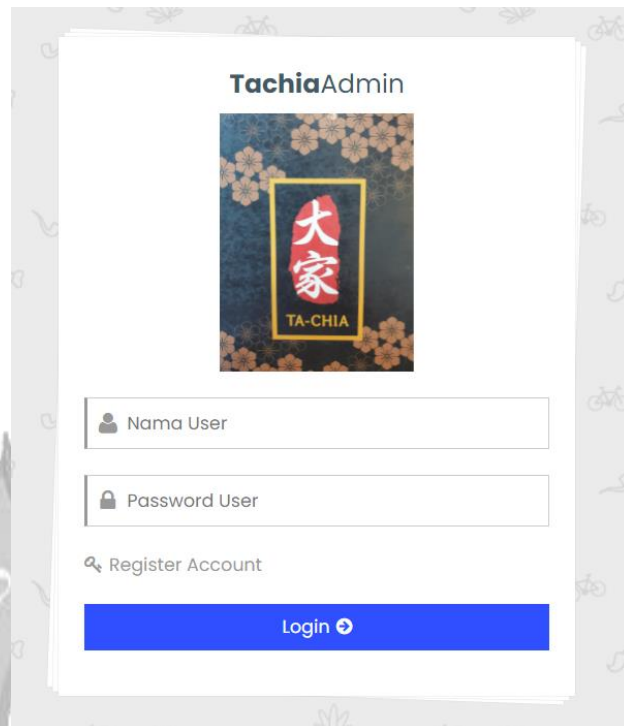
Pada *website* pencatatan keuangan dan inventori Restoran Ta Chia, terdapat 3 kategori pengguna, yaitu manajer, kasir, dan dapur. Untuk masuk ke dalam halaman *dashboard* pada masing-masing pihak, mereka harus melakukan *Log In* pada *Landing Page*. Berikut adalah kumpulan dari perancangan *user interface* sistem usulan.

4.4.2.1. Perancangan Lapisan Antarmuka (*Landing Page*)

Landing Page adalah bagian *website* yang akan ditampilkan pertama kali kepada para pengguna sistem. Halaman *landing page* pada sistem usulan terdiri dari

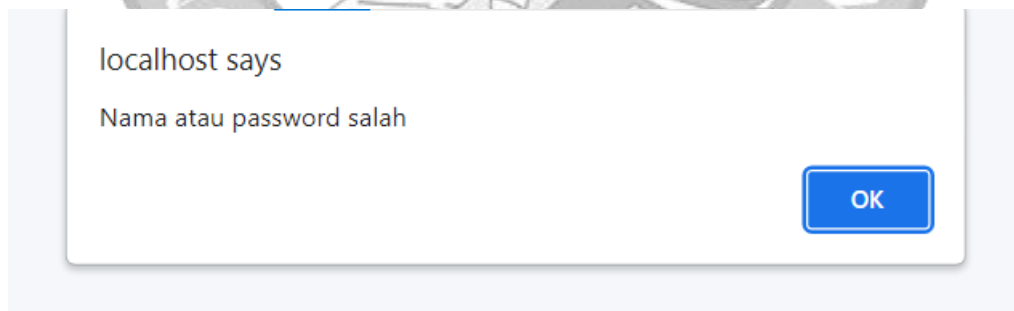
halaman *Log In* dan *register*. Berikut merupakan penjelasan dari masing-masing halaman.

1) Halaman *Log In*

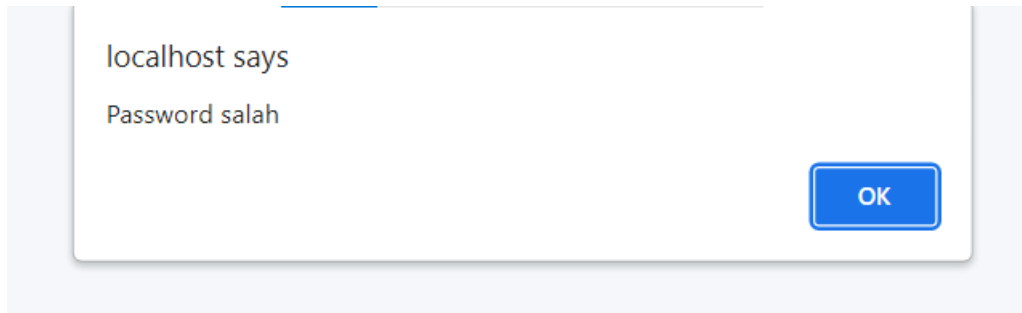


Gambar 4.78 Tampilan Halaman *Log In*

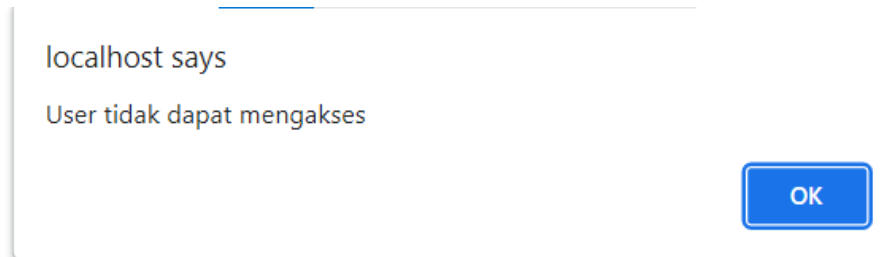
Pada halaman *Log In*, tersedia *form* untuk mengisi nama *user* dan *password user*. Dibawah kolom *field input* tersedia *link* untuk mengakses halaman *register user* baru. Tampilan halaman dapat dilihat pada Gambar 4.78.



Gambar 4.79 Tampilan notifikasi data *input* salah atau kolom kosong pada halaman *Log In*



Gambar 4.80 Tampilan notifikasi *user input* salah *password* pada halaman *Log In*



Gambar 4.81 Tampilan notifikasi *user* tidak dapat mengakses pada halaman *Log In*

Untuk mengakses halaman *dashboard* pada masing-masing pihak, pengguna perlu mengisi kolom yang tersedia dan menekan tombol "*Login*". Jika kolom masih kosong atau data yang diisi salah, maka sistem akan menampilkan notifikasi nama atau *password* salah. Jika nama yang diisi benar tetapi *password* salah maka sistem akan menampilkan notifikasi *password* salah. Jika status *user inactive*, maka sistem akan menampilkan notifikasi *user* tidak dapat mengakses. Setelah menekan tombol "*OK*" pada notifikasi, sistem akan balik menampilkan halaman *Log In*.

Jika status *user active*, maka sistem akan menampilkan halaman *dashboard* sesuai dengan *role* masing-masing *user*.

2) Halaman *Register*

TachiaAdmin
Register Account

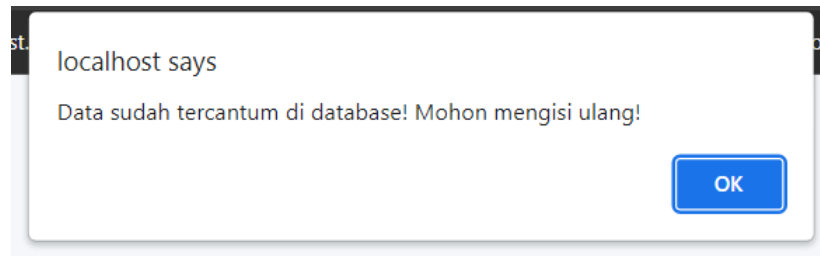
➔ Sudah punya account? [Login.](#)

Gambar 4.82 Tampilan Halaman *Sign Up*

Halaman *Sign Up* dapat diakses melalui halaman *Log In*. Pada halaman *Sign Up*, tersedia *form* untuk mengisi nama *user*, nomor telepon *user*, *password user*, dan *role user*. Isi pemilihan *role user* diambil dari *database role*. Pada bagian bawah *form* tersedia *link* untuk mengakses balik ke halaman *Log In*. Tampilan halaman dapat dilihat pada gambar di atas.

! Please fill out this field.

Gambar 4.83 Tampilan notifikasi kolom kosong pada halaman *Sign Up*



Gambar 4.84 Tampilan notifikasi data sudah ada pada halaman *Sign Up*

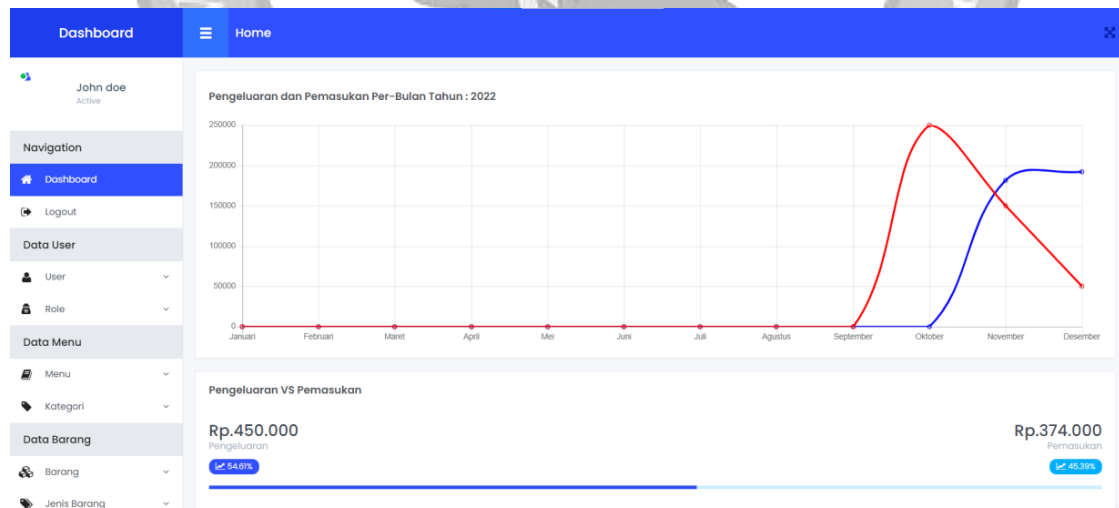
Untuk menambah data *user*, pengguna perlu mengisi kolom yang tersedia dan menekan tombol “*Submit*”. Jika kolom masih kosong, sistem akan menampilkan notifikasi pada bagian bawah kolom yang masih kosong seperti yang ditunjukkan pada Gambar 4.83. Jika data yang diisi sudah terdeteksi di dalam *database*, maka sistem akan menampilkan notifikasi yang ditunjukkan pada Gambar 4.84. Setelah menekan tombol “*OK*” pada notifikasi, sistem akan balik menampilkan halaman *Sign Up*.

Jika data yang diisi berhasil tersimpan ke dalam *database*, sistem akan menampilkan halaman *Log In*.

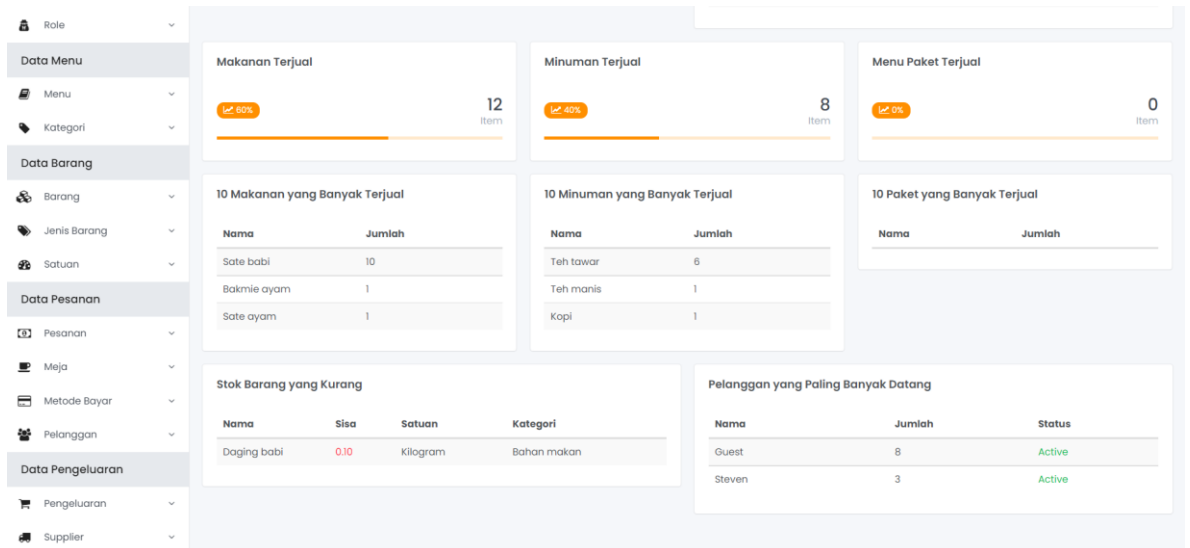
4.4.2.2. Perancangan Lapisan Antarmuka (Pihak Manajer)

Berikut merupakan tampilan halaman yang dapat diakses oleh pihak manajer.

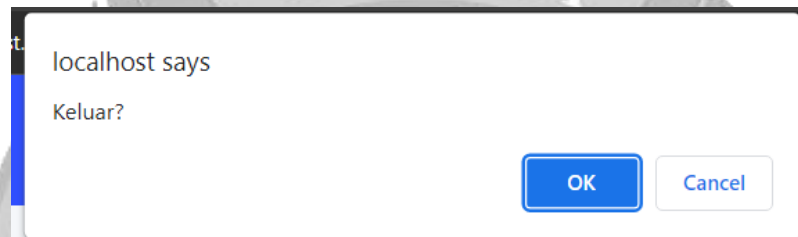
1) Halaman *Dashboard* Manajer



Gambar 4.85 Tampilan Halaman *Dashboard* Manajer (1)



Gambar 4.86 Tampilan Halaman *Dashboard* Manajer (2)

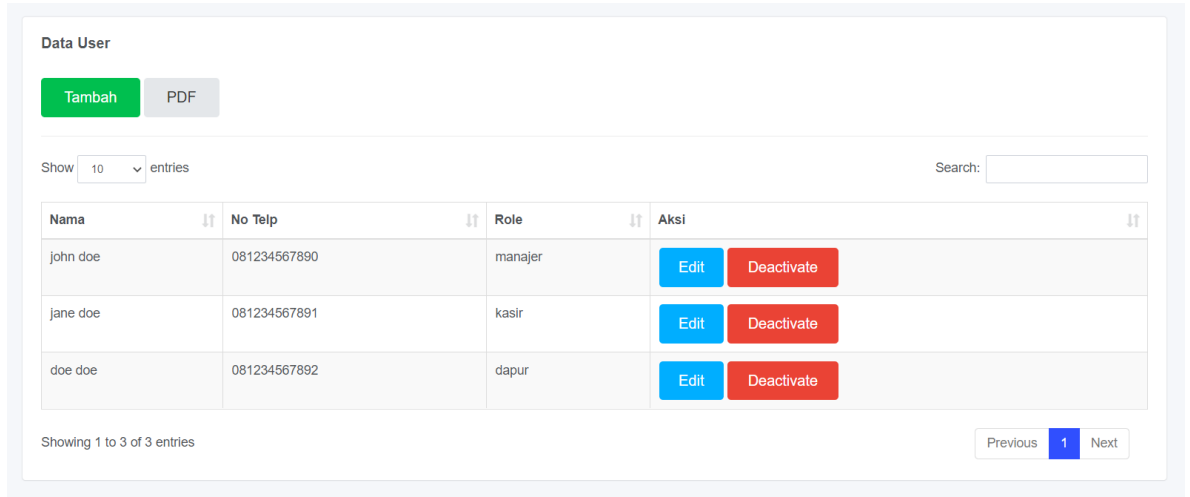


Gambar 4.87 Tampilan notifikasi konfirmasi *log out*

Pada halaman *dashboard*, tersedia beberapa kolom yang menampilkan informasi. Informasi yang ditampilkan adalah grafik perbandingan pengeluaran dan pemasukan per-bulan, jumlah perbandingan pengeluaran dan pemasukan, jumlah makanan terjual, jumlah minuman terjual, jumlah *menu* paket terjual, *top 10* makanan terjual, *top 10* minuman terjual, *top 10 menu* paket terjual, stok barang yang kurang, dan pelanggan yang paling banyak datang makan.

Pada sebelah kiri halaman terdapat *navigation bar* yang menunjukkan tombol akses halaman-halaman *master data*. Selain itu pada bagian atas *bar* terdapat tombol untuk melakukan *logout*. Setelah menekan tombol *logout*, sistem akan menampilkan notifikasi konfirmasi *logout*.

2) Halaman *Master Data User*



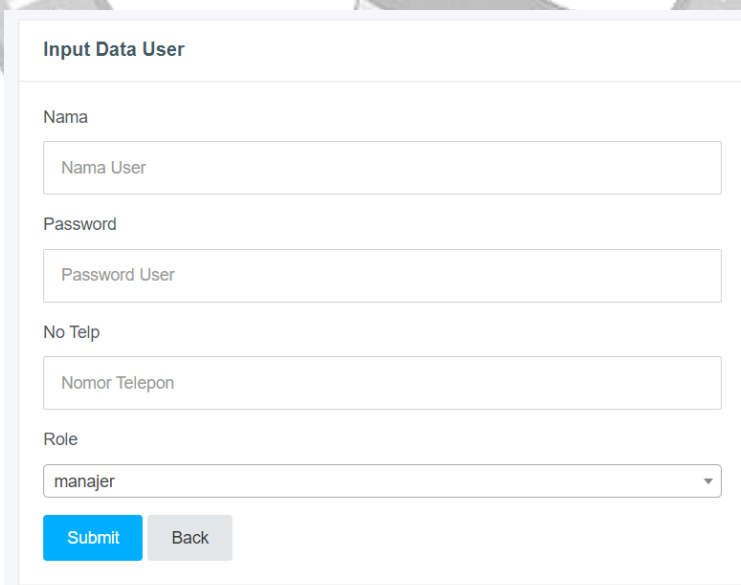
The screenshot shows a web interface titled "Data User". At the top left, there are two buttons: "Tambah" (Add) in green and "PDF" in grey. Below these is a "Show 10 entries" dropdown and a "Search:" input field. The main content is a table with four columns: "Nama", "No Telp", "Role", and "Aksi". The table contains three rows of user data. Each row has "Edit" and "Deactivate" buttons in the "Aksi" column. At the bottom left, it says "Showing 1 to 3 of 3 entries". At the bottom right, there are "Previous", "1", and "Next" navigation buttons.

Nama	No Telp	Role	Aksi
john doe	081234567890	manajer	Edit Deactivate
jane doe	081234567891	kasir	Edit Deactivate
doe doe	081234567892	dapur	Edit Deactivate

Gambar 4.88 Tampilan Halaman *Master Data User*

Halaman *Master Data User* adalah halaman yang menampilkan *list* data setiap *user* yang dapat menggunakan sistem usulan dalam bentuk tabel. Halaman hanya dapat diakses oleh pihak manajer. Data yang ditampilkan adalah nama *user*, nomor telepon *user*, dan *role user*.

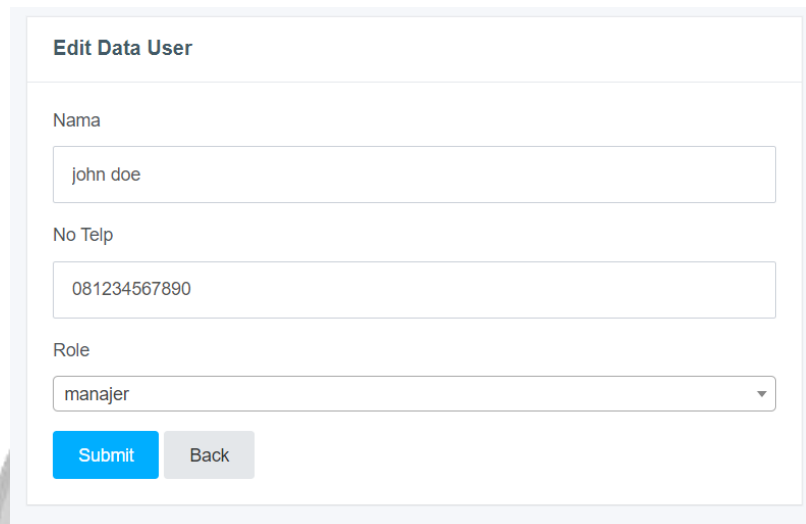
Pada bagian atas halaman telah tersedia dua tombol untuk mengakses halaman tambah data dan halaman PDF. Pada bagian kanan tabel terdapat kolom aksi yang menyediakan tombol untuk mengakses halaman *edit* data dan *deactivate* data. Tampilan halaman dapat dilihat pada gambar di atas.



The screenshot shows a form titled "Input Data User". It contains four input fields: "Nama" (with placeholder "Nama User"), "Password" (with placeholder "Password User"), "No Telp" (with placeholder "Nomor Telepon"), and "Role" (a dropdown menu with "manajer" selected). At the bottom, there are "Submit" and "Back" buttons.

Gambar 4.89 Tampilan Halaman Menambah Data *User*

Halaman *input* data *user* dapat diakses dengan menekan tombol “Tambah”. Untuk menambah data *user*, tersedia *form* untuk mengisi nama *user*, *password user*, nomor telepon *user*, dan *role user*. Tersedia tombol “Back” untuk balik ke halaman *master data* dan tombol “Submit” untuk menyimpan data ke dalam *database*. Tampilan halaman dapat dilihat pada gambar di atas.



Edit Data User

Nama
john doe

No Telp
081234567890

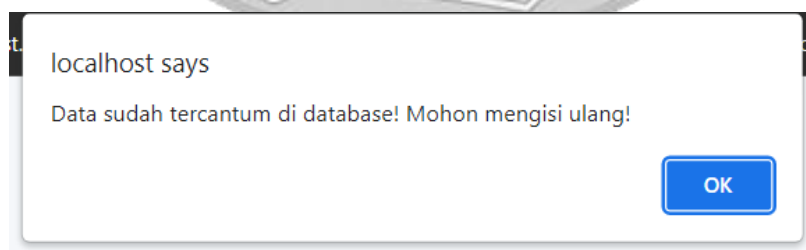
Role
manajer

Submit Back

Gambar 4.90 Tampilan Halaman Mengubah Data *User*

Halaman *edit* data *user* dapat diakses dengan menekan tombol “Edit”. Untuk mengubah data *user*, tersedia *form* untuk mengubah nama *user*, nomor telepon, dan *role user*. Tersedia tombol “Back” untuk balik ke halaman *master data* dan tombol “Submit” untuk menyimpan data ke dalam *database*.

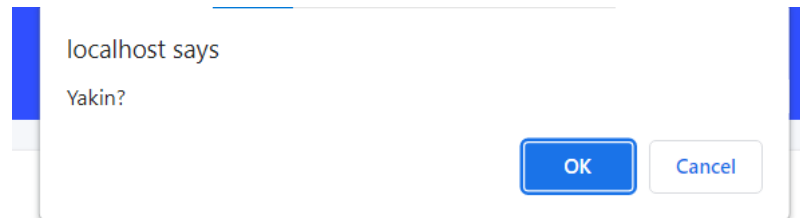
Ketika pengguna mengubah data diri sendiri, sistem akan menampilkan notifikasi “Mohon *log in* ulang” dan menampilkan kembali halaman *login*. Tampilan halaman dapat dilihat pada gambar di atas.



Gambar 4.91 Tampilan notifikasi data sudah ada pada halaman *master data user*

Setelah menekan tombol “*Submit*” setelah menambah atau mengubah data, jika data sudah tercantum di dalam *database*, maka sistem akan menampilkan notifikasi yang ditunjukkan pada gambar di atas.

Jika data yang berhasil tersimpan ke dalam *database*, sistem akan menampilkan halaman *Master Data User*.



Gambar 4.92 Tampilan notifikasi konfirmasi hapus data *user*

Setelah menekan tombol “*Deactivate*”, sistem akan menampilkan notifikasi konfirmasi hapus data. Jika ingin menghapus data, maka tekan tombol “OK”.

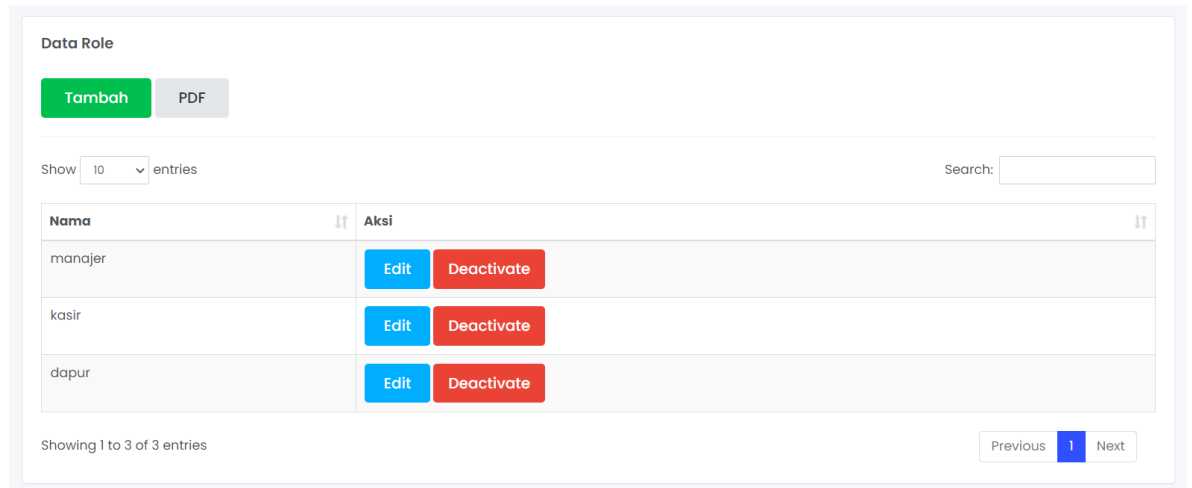
Data User Tachia

Nama	Nomor Telepon	Role
john doe	081234567890	manajer
jane doe	081234567891	kasir
doe doe	081234567892	dapur

Gambar 4.93 Tampilan PDF *Master Data User*

Halaman PDF dapat diakses dengan menekan tombol “PDF”. Tampilan halaman dapat dilihat pada gambar di atas.

3) Halaman *Master Data Role*



Gambar 4.94 Tampilan Halaman *Master Data Role*

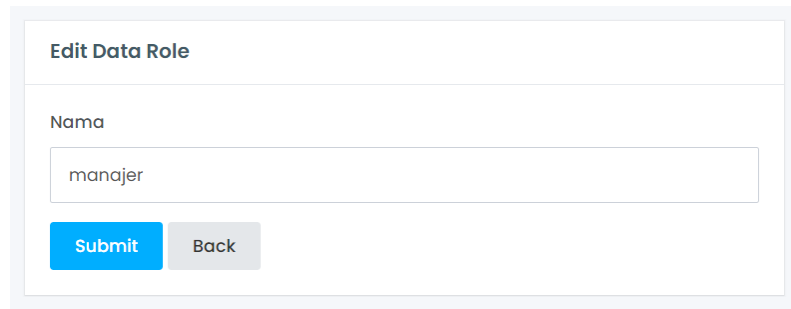
Halaman *Master Data Role* adalah halaman yang menampilkan *list* data setiap *role* dalam bentuk tabel. Halaman hanya dapat diakses oleh pihak manajer. Data yang ditampilkan adalah nama *role*.

Pada bagian atas halaman telah tersedia dua tombol untuk mengakses halaman tambah data dan halaman PDF. Pada bagian kanan tabel terdapat kolom aksi yang menyediakan tombol untuk mengakses halaman *edit* data dan *deactivate* data. Tampilan halaman dapat dilihat pada gambar di atas.

The screenshot shows a form titled "Input Data Role". It has a "Nama" label and a text input field with the placeholder text "Nama Role". Below the input field are two buttons: "Submit" (blue) and "Back" (grey).

Gambar 4.95 Tampilan Halaman Menambah Data *Role*

Halaman *input data role* dapat diakses dengan menekan tombol "Tambah". Untuk menambah data *user*, tersedia *form* untuk mengisi nama *role*. Tersedia tombol "Back" untuk balik ke halaman *master data* dan tombol "Submit" untuk menyimpan data ke dalam *database*. Tampilan halaman dapat dilihat pada gambar di atas.



Edit Data Role

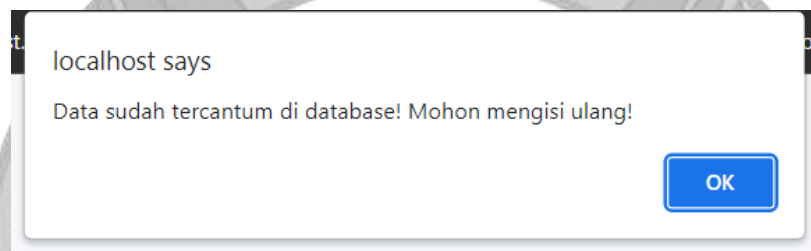
Nama

manajer

Submit Back

Gambar 4.96 Tampilan Halaman Mengubah Data *Role*

Halaman *edit data role* dapat diakses dengan menekan tombol “*Edit*”. Untuk mengubah data *role*, tersedia *form* untuk mengubah nama *role*. Tersedia tombol “*Back*” untuk balik ke halaman *master data* dan tombol “*Submit*” untuk menyimpan data ke dalam *database*. Tampilan halaman dapat dilihat pada di atas.



Gambar 4.97 Tampilan notifikasi data sudah ada pada halaman *master data role*

Setelah menekan tombol “*Submit*” setelah menambah atau mengubah data, jika data sudah tercantum di dalam *database*, maka sistem akan menampilkan notifikasi yang ditunjukkan pada gambar di atas.

Jika data yang berhasil tersimpan ke dalam *database*, sistem akan menampilkan halaman *Master Data Role*.



Gambar 4.98 Tampilan notifikasi konfirmasi hapus data *role*

Setelah menekan tombol “*Deactivate*”, sistem akan menampilkan notifikasi hapus data. Jika ingin menghapus data, maka tekan tombol “*OK*”.

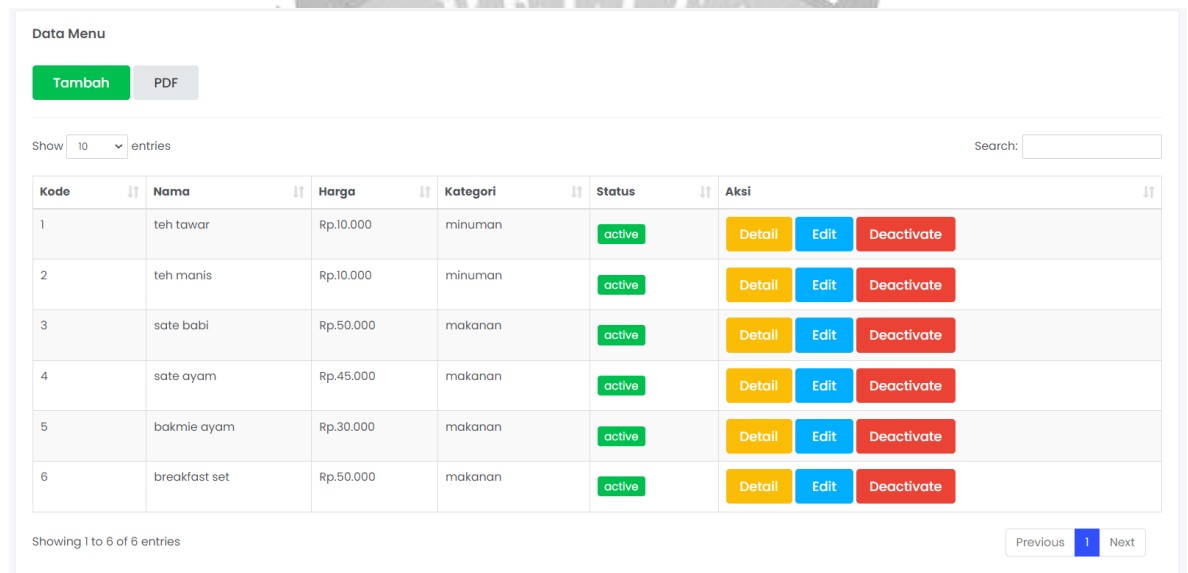
Data Role Tachia

Nama
manajer
kasir
dapur

Gambar 4.99 Tampilan PDF *Master Data Role*

Halaman PDF dapat diakses dengan menekan tombol “PDF”. Tampilan halaman dapat dilihat pada gambar di atas.

4) Halaman *Master Data Menu*



The screenshot shows a web interface for 'Data Menu'. At the top, there are two buttons: 'Tambah' (green) and 'PDF' (grey). Below the buttons, there is a 'Show 10 entries' dropdown and a search box. The main content is a table with 6 rows and 7 columns: Kode, Nama, Harga, Kategori, Status, and Aksi. Each row contains menu details and three action buttons: Detail (yellow), Edit (blue), and Deactivate (red). The status for all items is 'active'.

Kode	Nama	Harga	Kategori	Status	Aksi
1	teh tawar	Rp.10.000	minuman	active	Detail Edit Deactivate
2	teh manis	Rp.10.000	minuman	active	Detail Edit Deactivate
3	sate babi	Rp.50.000	makanan	active	Detail Edit Deactivate
4	sate ayam	Rp.45.000	makanan	active	Detail Edit Deactivate
5	bakmie ayam	Rp.30.000	makanan	active	Detail Edit Deactivate
6	breakfast set	Rp.50.000	makanan	active	Detail Edit Deactivate

Showing 1 to 6 of 6 entries

Previous 1 Next

Gambar 4.100 Tampilan Halaman *Master Data Menu*

Halaman *Master Data Menu* adalah halaman yang menampilkan *list* data setiap *menu* dalam bentuk tabel. Halaman hanya dapat diakses oleh pihak manajer. Data yang ditampilkan adalah kode *menu*, nama *menu*, harga *menu*, kategori *menu*, dan status *menu*. Isi status *menu* terdiri dari *active* dan *inactive*.

Pada bagian atas halaman telah tersedia dua tombol untuk mengakses halaman tambah data dan halaman PDF. Pada bagian kanan tabel terdapat kolom aksi yang menyediakan tombol untuk mengakses halaman data *detail menu*, *edit* data, dan *deactivate* data. Tampilan halaman dapat dilihat pada gambar di atas.

Input Data Menu

Kode

Nama

Harga

Kategori

Gambar 4.101 Tampilan Halaman Menambah Data *Menu*

Halaman *input data menu* dapat diakses dengan menekan tombol “Tambah”. Untuk menambah data *user*, tersedia *form* untuk mengisi kode *menu*, nama *menu*, harga *menu*, dan kategori *menu*. Tersedia tombol “Back” untuk balik ke halaman *master data* dan tombol “Submit” untuk menyimpan data ke dalam *database*. Tampilan halaman dapat dilihat pada gambar di atas.

Edit Data Menu

Kode

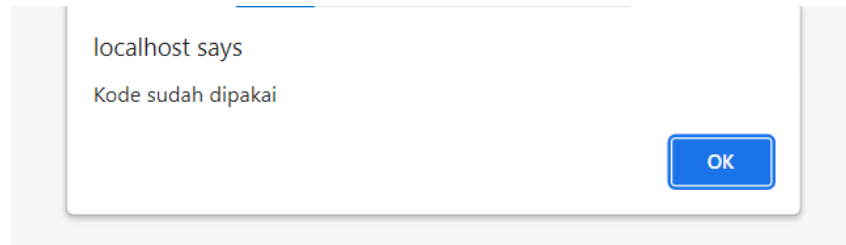
Nama

Harga

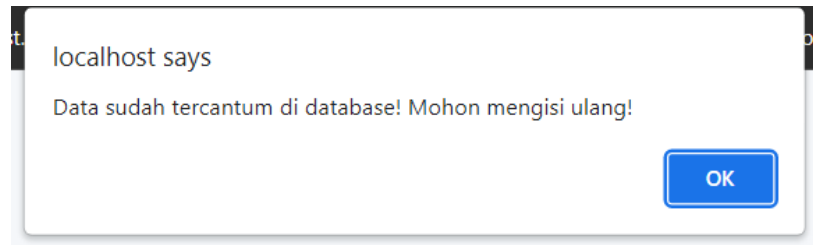
Kategori

Gambar 4.102 Tampilan Halaman Mengubah Data *Menu*

Halaman *edit data user* dapat diakses dengan menekan tombol “Edit”. Untuk mengubah data *menu*, tersedia *form* untuk mengubah kode *menu*, nama *menu*, harga *menu*, dan kategori *menu*. Tersedia tombol “Back” untuk balik ke halaman *master data* dan tombol “Submit” untuk menyimpan data ke dalam *database*. Tampilan halaman dapat dilihat pada gambar di atas.



Gambar 4.103 Tampilan notifikasi kode sudah ada pada halaman *master data menu*



Gambar 4.104 Tampilan notifikasi data sudah ada pada halaman *master data menu*

Setelah menekan tombol “*Submit*” setelah menambah dan mengubah data, jika data sudah tercantum di dalam *database*, maka sistem akan menampilkan notifikasi yang ditunjukkan pada gambar di atas. Jika kode yang diisi sudah terpakai oleh *menu* lain, maka sistem akan menampilkan notifikasi kode sudah dipakai.

Jika data yang berhasil tersimpan ke dalam *database*, sistem akan menampilkan halaman *Master Data Menu*.

Kategori	Status	Aksi
minuman	inactive	Detail Edit Activate
minuman	active	Detail Edit Deactivate
makanan	active	Detail Edit Deactivate

Gambar 4.105 Tampilan status data *menu inactive*

Setelah menekan tombol “*Deactivate*”, status *menu* akan berubah menjadi *inactive* seperti yang ditunjukkan pada gambar di atas. Jika ingin mengaktifasi kembali, maka tekan tombol “*Activate*”.

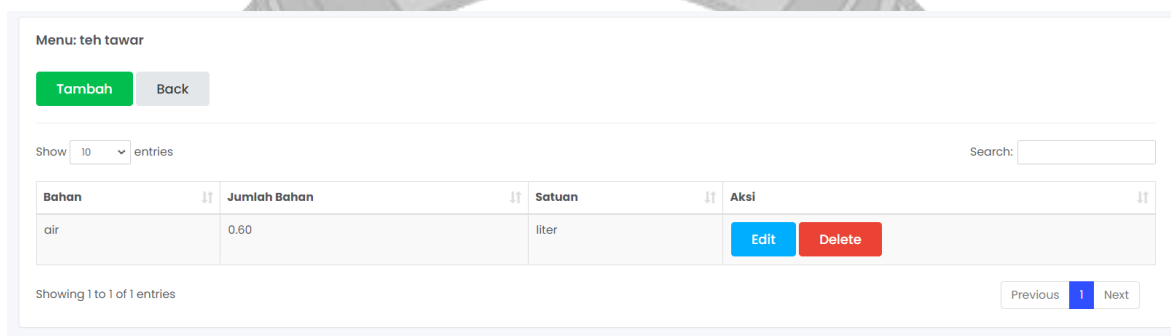
Data Menu Tachia

Kode	Nama	Harga	Kategori
1	teh tawar	Rp.10.000	minuman
2	teh manis	Rp.10.000	minuman
3	sate babi	Rp.50.000	makanan
4	sate ayam	Rp.45.000	makanan
5	bakmie ayam	Rp.30.000	makanan
6	breakfast set	Rp.50.000	makanan

Gambar 4.106 Tampilan PDF *Master Data Menu*

Halaman PDF dapat diakses dengan menekan tombol “PDF”. Tampilan halaman dapat dilihat pada gambar di atas.

5) Halaman *Master Data Detail Menu*

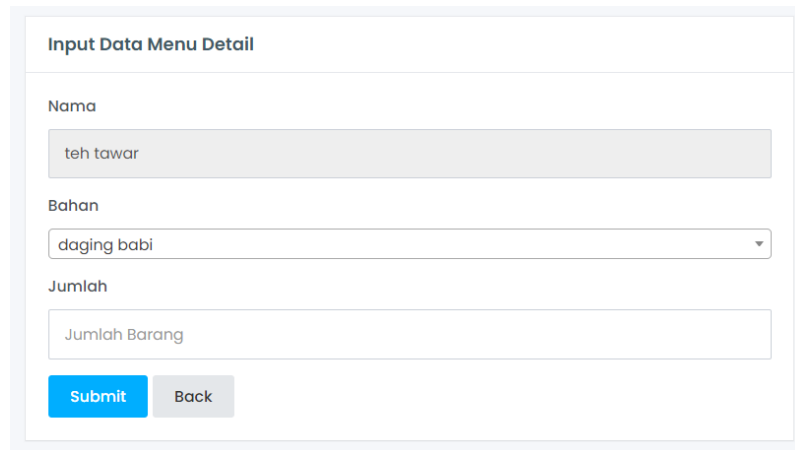


The screenshot shows the 'Master Data Detail Menu' page for the menu item 'teh tawar'. At the top, there are two buttons: 'Tambah' (Add) in green and 'Back' in grey. Below this, there is a 'Show 10 entries' dropdown menu and a search box. The main content is a table with the following columns: 'Bahan' (Ingredient), 'Jumlah Bahan' (Quantity), 'Satuan' (Unit), and 'Aksi' (Action). The table contains one entry: 'air' with a quantity of '0.60' and unit 'liter'. The 'Aksi' column has two buttons: 'Edit' in blue and 'Delete' in red. At the bottom, there is a 'Showing 1 to 1 of 1 entries' indicator and a pagination control with 'Previous', '1', and 'Next' buttons.

Gambar 4.107 Tampilan Halaman *Master Data Detail Menu*

Halaman *Master Data Detail Menu* adalah halaman yang menampilkan *list detail* data setiap *menu*. Halaman hanya dapat diakses oleh pihak manajer dengan menekan tombol “*Detail*” pada halaman *Master Data Menu*. Data yang ditampilkan adalah nama barang, jumlah bahan, dan satuan barang.

Pada bagian atas halaman telah tersedia dua tombol untuk mengakses halaman tambah data dan halaman PDF. Pada bagian kanan tabel terdapat kolom aksi yang menyediakan tombol untuk mengakses halaman *edit* data dan *delete* data. Tampilan halaman dapat dilihat pada gambar di atas.



Input Data Menu Detail

Nama
teh tawar

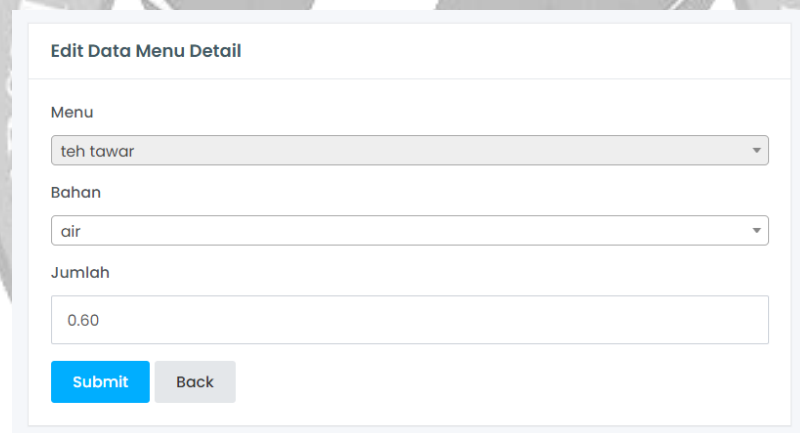
Bahan
daging babi

Jumlah
Jumlah Barang

Submit Back

Gambar 4.108 Tampilan Halaman Menambah Data *Detail Menu*

Halaman input data menu dapat diakses dengan menekan tombol “Tambah”. Untuk menambah data *menu*, tersedia *form* untuk mengisi nama bahan dan jumlah barang. Tersedia tombol “Back” untuk balik ke halaman *master data* dan tombol “Submit” untuk menyimpan data ke dalam *database*. Tampilan halaman dapat dilihat pada gambar di atas.



Edit Data Menu Detail

Menu
teh tawar

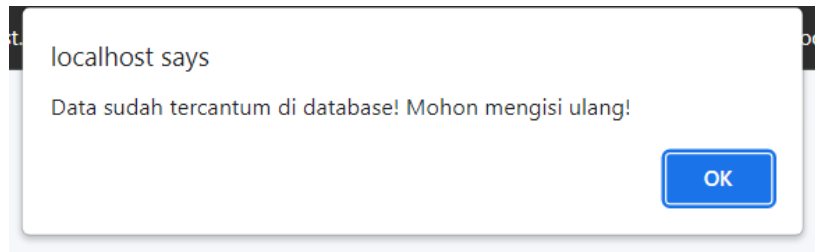
Bahan
air

Jumlah
0.60

Submit Back

Gambar 4.109 Tampilan Halaman *Edit Data Detail Menu*

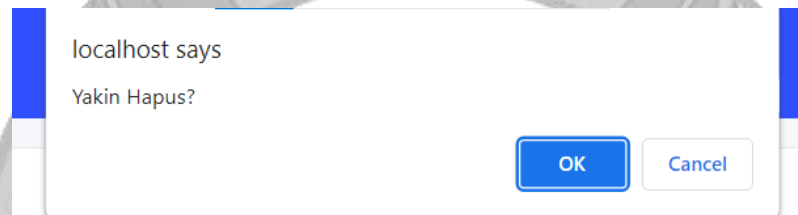
Halaman *edit data menu* dapat diakses dengan menekan tombol “Edit”. Untuk mengubah data *menu*, tersedia *form* untuk mengubah nama bahan dan jumlah barang. Tersedia tombol “Back” untuk balik ke halaman *master data* dan tombol “Submit” untuk menyimpan data ke dalam *database*. Tampilan halaman dapat dilihat pada gambar di atas.



Gambar 4.110 Tampilan notifikasi data sudah ada pada halaman *master data detail menu*

Setelah menekan tombol “*Submit*” setelah menambah dan mengubah data, jika data sudah tercantum di dalam *database*, maka sistem akan menampilkan notifikasi yang ditunjukkan pada gambar di atas.

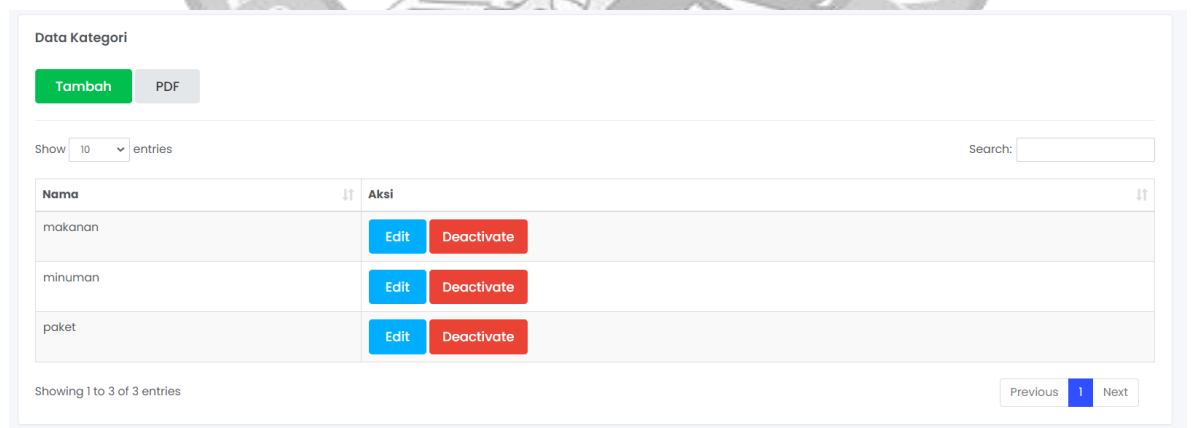
Jika data yang berhasil tersimpan ke dalam *database*, sistem akan menampilkan halaman *Master Data Detail Menu*.



Gambar 4.111 Tampilan notifikasi konfirmasi hapus data *detail menu*

Setelah menekan tombol “*Deactivate*”, sistem akan menampilkan notifikasi konfirmasi hapus data. Jika ingin menghapus data, maka tekan tombol “OK”.

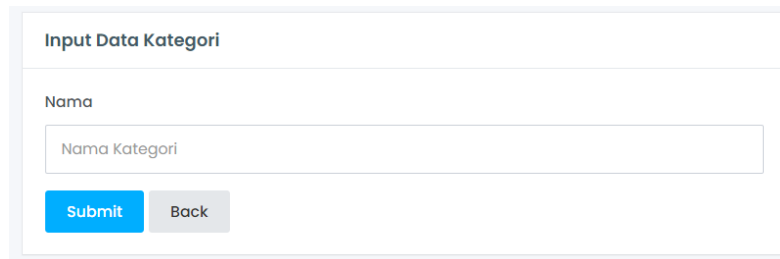
6) Halaman *Master Data Kategori*



Gambar 4.112 Tampilan Halaman *Master Data Kategori*

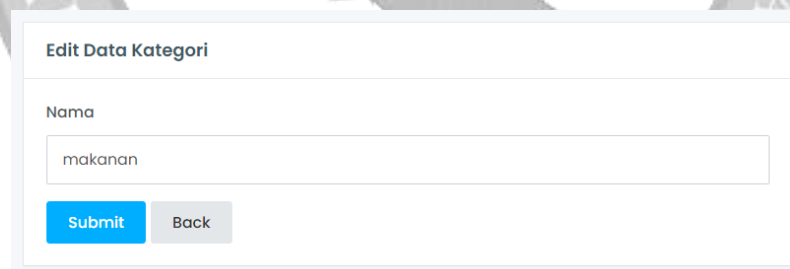
Halaman *Master Data* Kategori adalah halaman yang menampilkan *list* data setiap kategori dalam bentuk tabel. Halaman hanya dapat diakses oleh pihak manajer. Data yang ditampilkan adalah nama kategori.

Pada bagian atas halaman telah tersedia dua tombol untuk mengakses halaman tambah data dan halaman PDF. Pada bagian kanan tabel terdapat kolom aksi yang menyediakan tombol untuk mengakses halaman *edit* data dan *deactivate* data. Tampilan halaman dapat dilihat pada gambar di atas.



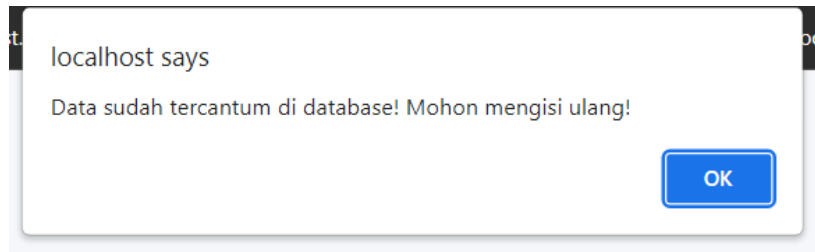
Gambar 4.113 Tampilan Halaman Menambah Data Kategori

Halaman *input data* kategori dapat diakses dengan menekan tombol “Tambah”. Untuk menambah data kategori, tersedia *form* untuk mengisi nama kategori. Tersedia tombol “Back” untuk balik ke halaman *master data* dan tombol “Submit” untuk menyimpan data ke dalam *database*. Tampilan halaman dapat dilihat pada gambar di atas.



Gambar 4.114 Tampilan Halaman Mengubah Data Kategori

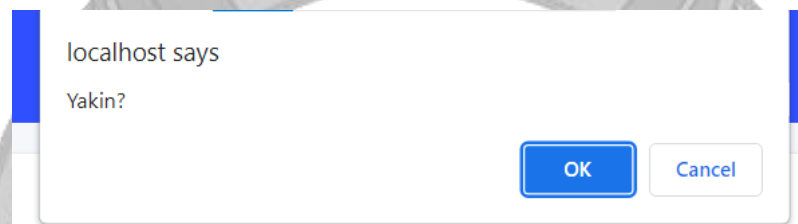
Halaman *edit data* kategori dapat diakses dengan menekan tombol “Edit”. Untuk mengubah data kategori, tersedia *form* untuk mengubah nama kategori. Tersedia tombol “Back” untuk balik ke halaman *master data* dan tombol “Submit” untuk menyimpan data ke dalam *database*. Tampilan halaman dapat dilihat pada gambar di atas.



Gambar 4.115 Tampilan notifikasi data sudah ada pada halaman *master data* kategori

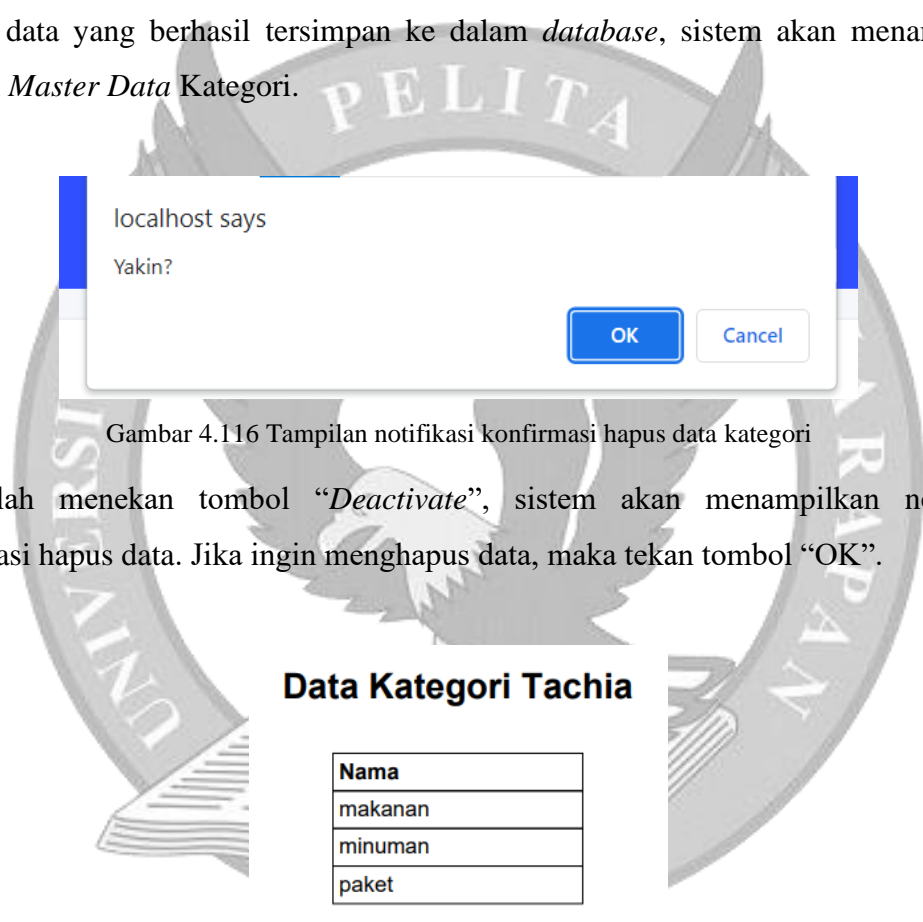
Setelah menekan tombol “*Submit*” setelah menambah dan mengubah data, jika data sudah tercantum di dalam *database*, maka sistem akan menampilkan notifikasi yang ditunjukkan pada gambar di atas.

Jika data yang berhasil tersimpan ke dalam *database*, sistem akan menampilkan halaman *Master Data* Kategori.



Gambar 4.116 Tampilan notifikasi konfirmasi hapus data kategori

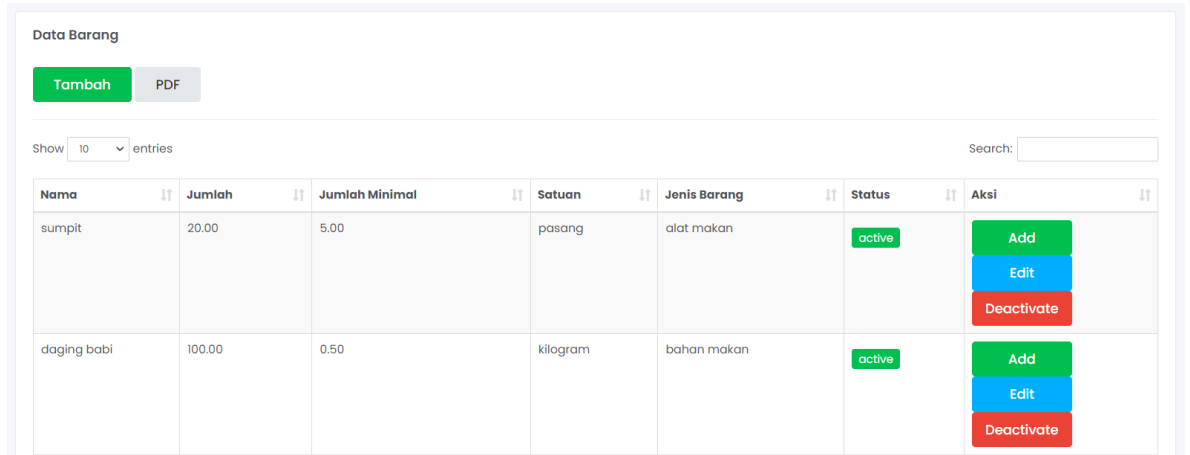
Setelah menekan tombol “*Deactivate*”, sistem akan menampilkan notifikasi konfirmasi hapus data. Jika ingin menghapus data, maka tekan tombol “OK”.



Gambar 4.117 Tampilan PDF *Master Data* Kategori

Halaman PDF dapat diakses dengan menekan tombol “PDF”. Tampilan halaman dapat dilihat pada gambar di atas.

7) Halaman *Master Data* Barang



Data Barang

Tambah PDF

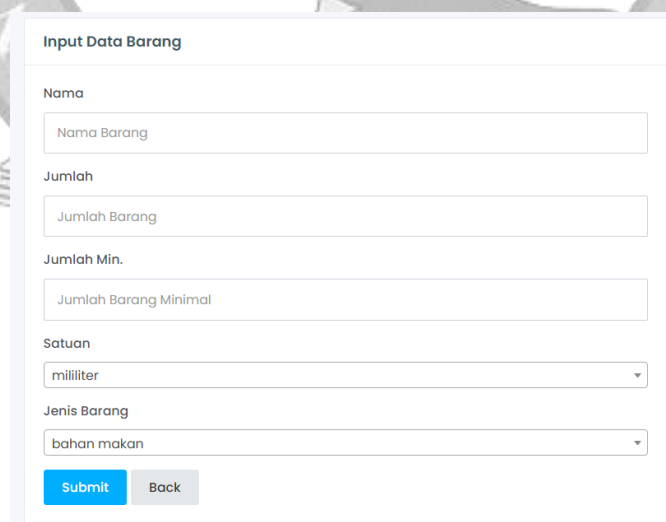
Show 10 entries Search:

Nama	Jumlah	Jumlah Minimal	Satuan	Jenis Barang	Status	Aksi
sumpit	20.00	5.00	pasang	alat makan	active	Add Edit Deactivate
daging babi	100.00	0.50	kilogram	bahan makan	active	Add Edit Deactivate

Gambar 4.118 Tampilan Halaman *Master Data* Barang

Halaman *Master Data* Barang adalah halaman yang menampilkan *list* data setiap barang dalam bentuk tabel. Halaman hanya dapat diakses oleh pihak manajer. Data yang ditampilkan adalah nama barang, jumlah barang, jumlah minimal barang, satuan barang, jenis barang, dan status barang. Isi dari status barang terdiri dari *active* dan *inactive*.

Pada bagian atas halaman telah tersedia dua tombol untuk mengakses halaman tambah data dan halaman PDF. Pada bagian kanan tabel terdapat kolom aksi yang menyediakan tombol untuk mengakses halaman tambah jumlah bahan, *edit* data, dan *deactivate* data. Tampilan halaman dapat dilihat pada gambar di atas.



Input Data Barang

Nama

Jumlah

Jumlah Min.

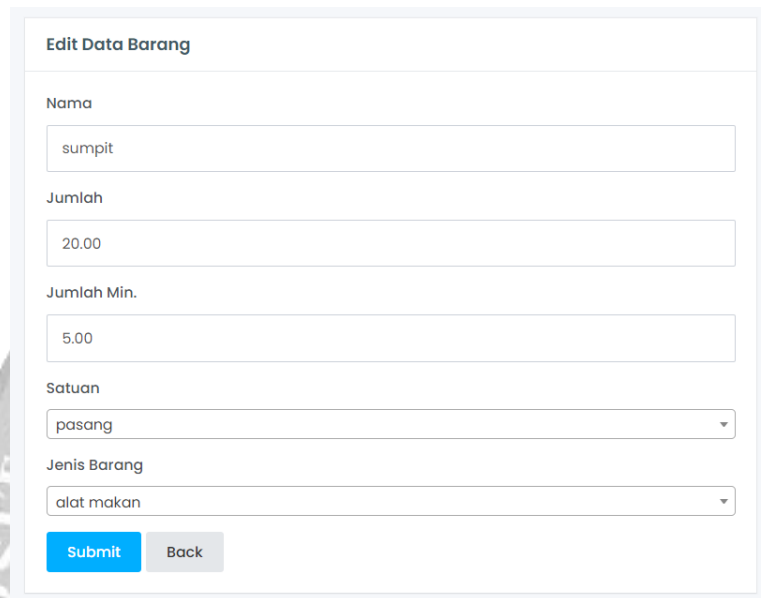
Satuan

Jenis Barang

[Submit](#) [Back](#)

Gambar 4.119 Tampilan Halaman Menambah Data Barang

Halaman *input data* barang dapat diakses dengan menekan tombol “Tambah”. Untuk menambah data barang, tersedia *form* untuk mengisi nama barang, jumlah barang, jumlah minimal barang, satuan barang, jenis barang, dan status barang. Tersedia tombol “Back” untuk balik ke halaman *master data* dan tombol “Submit” untuk menyimpan data ke dalam *database*. Tampilan halaman dapat dilihat pada gambar di atas.

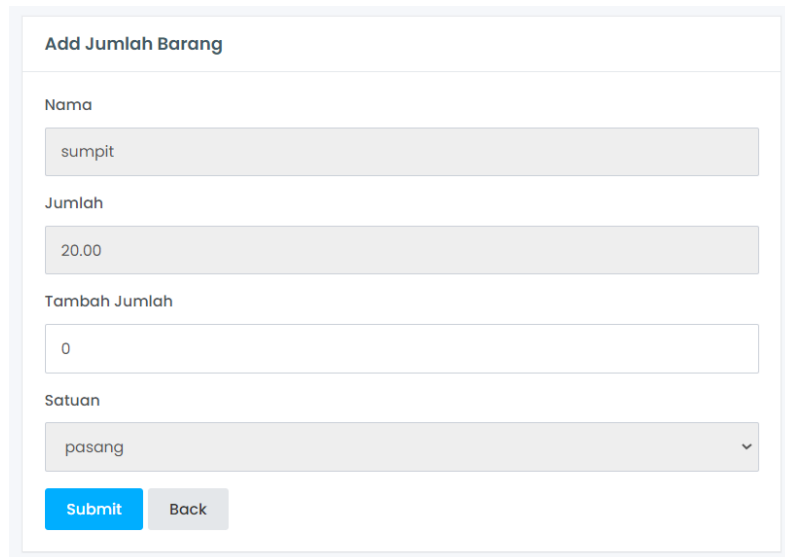


The image shows a web form titled "Edit Data Barang". It contains the following fields and controls:

- Nama:** A text input field containing the value "sumpit".
- Jumlah:** A text input field containing the value "20.00".
- Jumlah Min.:** A text input field containing the value "5.00".
- Satuan:** A dropdown menu with "pasang" selected.
- Jenis Barang:** A dropdown menu with "alat makan" selected.
- Buttons:** A blue "Submit" button and a grey "Back" button.

Gambar 4.120 Tampilan Halaman Mengubah Data Barang

Halaman *edit data* barang dapat diakses dengan menekan tombol “Edit”. Untuk mengubah data barang, tersedia *form* untuk mengubah nama barang, jumlah barang, jumlah minimal barang, satuan barang, jenis barang, dan status barang. Tersedia tombol “Back” untuk balik ke halaman *master data* dan tombol “Submit” untuk menyimpan data ke dalam *database*. Tampilan halaman dapat dilihat pada gambar di atas.



Add Jumlah Barang

Nama
sumpit

Jumlah
20.00

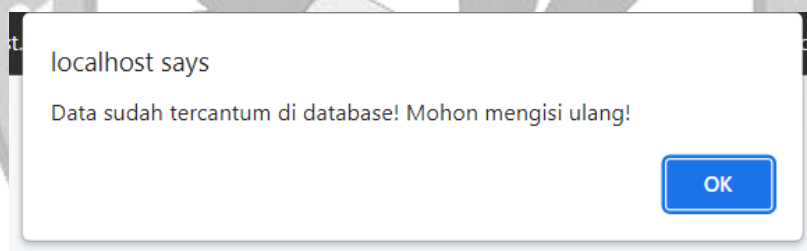
Tambah Jumlah
0

Satuan
pasang

Submit Back

Gambar 4.121 Tampilan Halaman Tambah Stok Barang

Halaman tambah data barang dapat diakses dengan menekan tombol “Add”. Untuk menambah jumlah barang, tersedia *form* untuk mengisi tambah jumlah barang. Tersedia tombol “Back” untuk balik ke halaman *master data* dan tombol “Submit” untuk menyimpan data ke dalam *database*. Tampilan halaman dapat dilihat pada gambar di atas.



Gambar 4.122 Tampilan notifikasi data sudah ada pada halaman *master data* barang

Setelah menekan tombol “Submit” setelah menambah dan mengubah data, jika data sudah tercantum di dalam *database*, maka sistem akan menampilkan notifikasi yang ditunjukkan pada gambar di atas.

Jika data yang berhasil tersimpan ke dalam *database*, sistem akan menampilkan halaman *Master Data* Barang.

Jenis Barang	Status	Aksi
alat makan	inactive	<input type="button" value="Add"/> <input type="button" value="Edit"/> <input type="button" value="Activate"/>

Gambar 4.123 Tampilan status data barang *inactive*

Setelah menekan tombol “*Deactivate*”, status barang akan berubah menjadi *inactive* seperti yang ditunjukkan pada gambar di atas. Jika ingin mengaktifasi kembali, maka tekan tombol “*Activate*”.

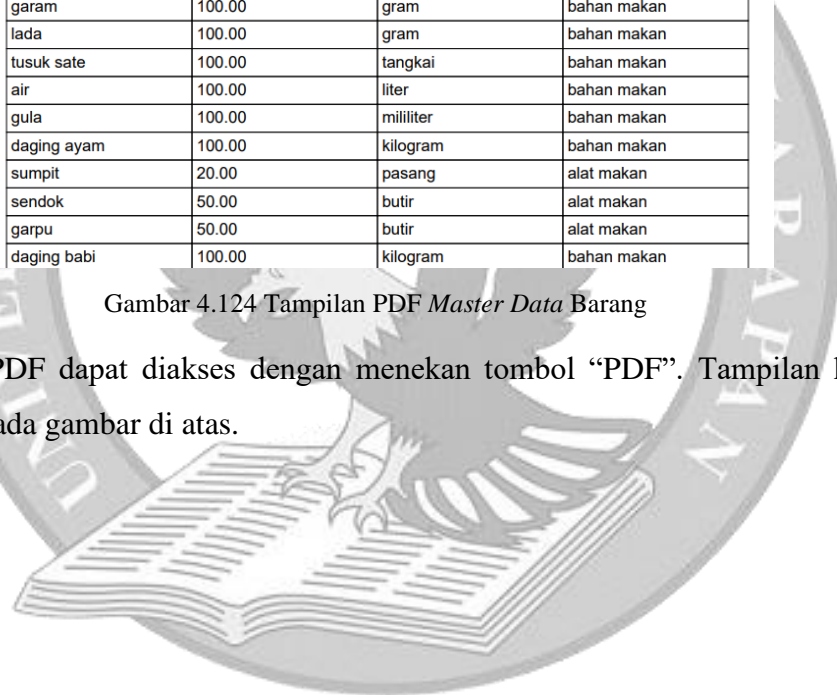


Data Barang Tachia

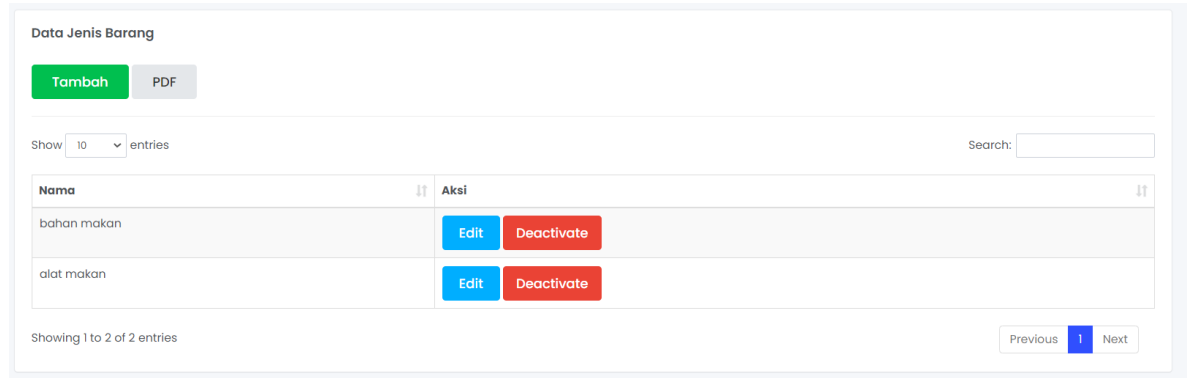
Nama	Jumlah	Satuan	Kategori
garam	100.00	gram	bahan makan
lada	100.00	gram	bahan makan
tusuk sate	100.00	tangkai	bahan makan
air	100.00	liter	bahan makan
gula	100.00	mililiter	bahan makan
daging ayam	100.00	kilogram	bahan makan
sumpit	20.00	pasang	alat makan
sendok	50.00	butir	alat makan
garpu	50.00	butir	alat makan
daging babi	100.00	kilogram	bahan makan

Gambar 4.124 Tampilan PDF *Master Data* Barang

Halaman PDF dapat diakses dengan menekan tombol “PDF”. Tampilan halaman dapat dilihat pada gambar di atas.



8) Halaman *Master Data* Jenis Barang



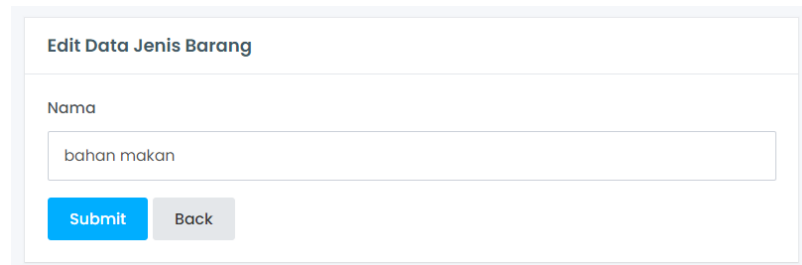
Gambar 4.125 Tampilan Halaman *Master Data* Jenis Barang

Halaman *Master Data* Jenis Barang adalah halaman yang menampilkan *list* data setiap jenis barang dalam bentuk tabel. Halaman hanya dapat diakses oleh pihak manajer. Data yang ditampilkan adalah nama jenis barang.

Pada bagian atas halaman telah tersedia dua tombol untuk mengakses halaman tambah data dan halaman *PDF*. Pada bagian kanan tabel terdapat kolom aksi yang menyediakan tombol untuk mengakses halaman *edit* data dan *deactivate* data. Tampilan halaman dapat dilihat pada gambar di atas.

Gambar 4.126 Tampilan Halaman Mengubah Data Jenis Barang

Halaman *input data* jenis barang dapat diakses dengan menekan tombol “Tambah”. Untuk menambah data jenis barang, tersedia *form* untuk mengisi nama jenis barang. Tersedia tombol “*Back*” untuk balik ke halaman *master data* dan tombol “*Submit*” untuk menyimpan data ke dalam *database*. Tampilan halaman dapat dilihat pada gambar di atas.



Edit Data Jenis Barang

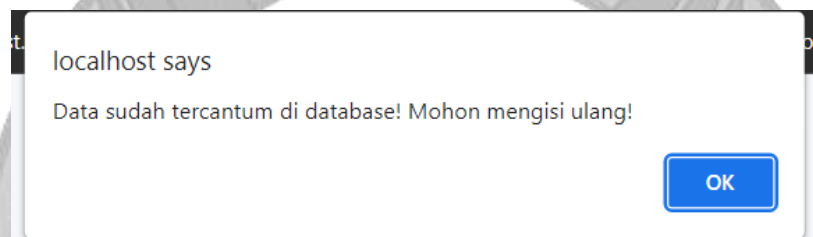
Nama

bahan makan

Submit Back

Gambar 4.127 Tampilan Halaman Mengubah Data Jenis Barang

Halaman *edit data* jenis barang dapat diakses dengan menekan tombol “*Edit*”. Untuk mengubah data jenis barang, tersedia *form* untuk mengubah nama jenis barang. Tersedia tombol “*Back*” untuk balik ke halaman *master data* dan tombol “*Submit*” untuk menyimpan data ke dalam *database*. Tampilan halaman dapat dilihat pada gambar di atas.



Gambar 4.128 Tampilan notifikasi data sudah ada pada halaman *master data* jenis barang

Setelah menekan tombol “*Submit*” setelah menambah dan mengubah data, jika data sudah tercantum di dalam *database*, maka sistem akan menampilkan notifikasi yang ditunjukkan pada gambar di atas.

Jika data yang berhasil tersimpan ke dalam *database*, sistem akan menampilkan halaman *Master Data* Jenis Barang.



Gambar 4.129 Tampilan notifikasi konfirmasi hapus data jenis barang

Setelah menekan tombol “*Deactivate*”, sistem akan menampilkan notifikasi konfirmasi hapus data. Jika ingin menghapus data, maka tekan tombol “OK”.

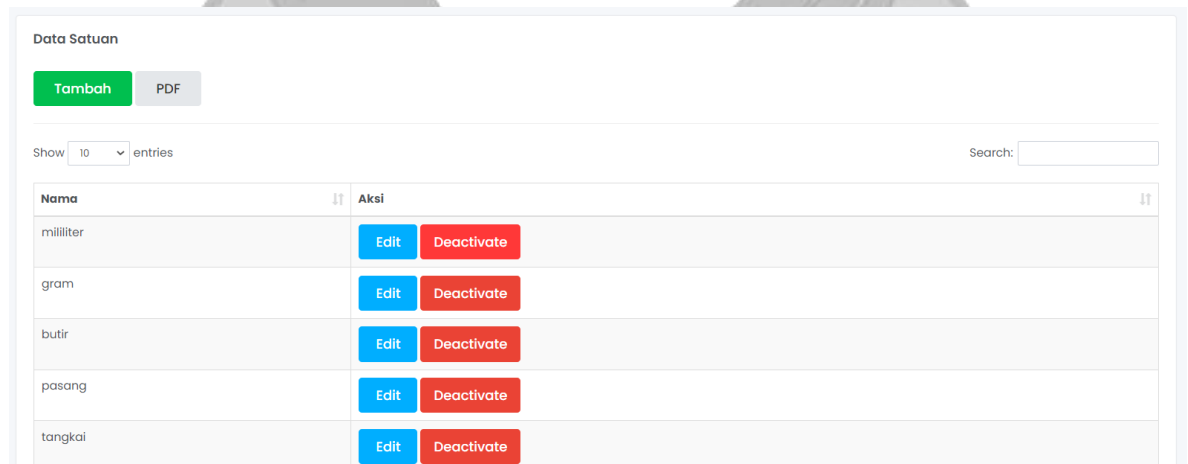
Data Jenis Barang Tachia

Nama
bahan makan
alat makan

Gambar 4.130 Tampilan PDF *Master Data* Jenis Barang

Halaman PDF dapat diakses dengan menekan tombol “PDF”. Tampilan halaman dapat dilihat pada gambar di atas.

9) Halaman *Master Data* Satuan

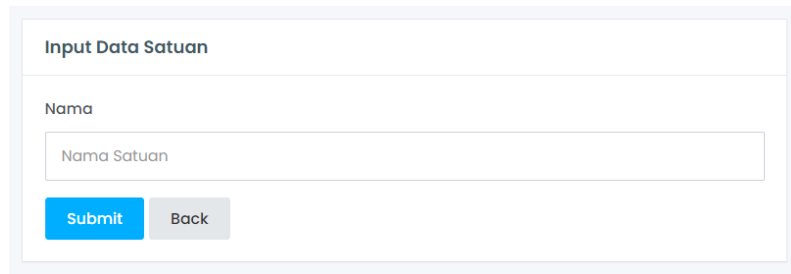


Nama	Aksi
milliliter	Edit Deactivate
gram	Edit Deactivate
butir	Edit Deactivate
pasang	Edit Deactivate
tangkai	Edit Deactivate

Gambar 4.131 Tampilan Halaman *Master Data* Satuan

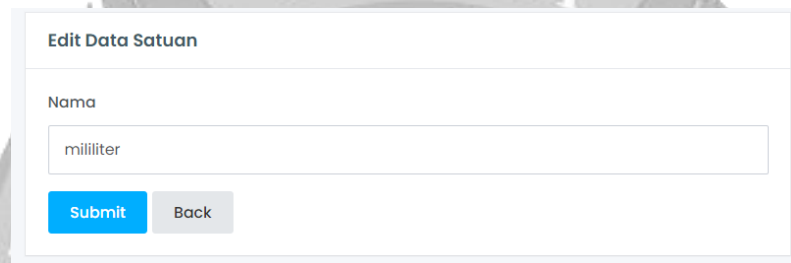
Halaman *Master Data* Satuan adalah halaman yang menampilkan *list* data setiap satuan dalam bentuk tabel. Halaman hanya dapat diakses oleh pihak manajer. Data yang ditampilkan adalah nama satuan.

Pada bagian atas halaman telah tersedia dua tombol untuk mengakses halaman tambah data dan halaman PDF. Pada bagian kanan tabel terdapat kolom aksi yang menyediakan tombol untuk mengakses halaman *edit* data dan *deactivate* data. Tampilan halaman dapat dilihat pada gambar di atas.



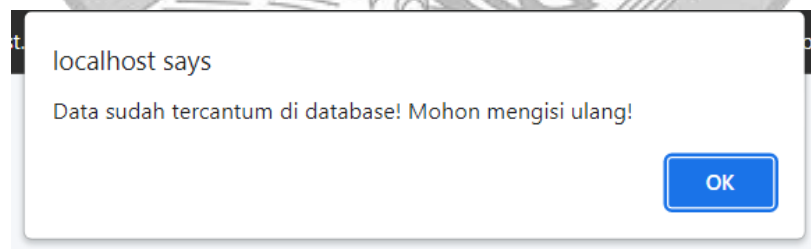
Gambar 4.132 Tampilan Halaman Mengubah Data Satuan

Halaman *input data* satuan dapat diakses dengan menekan tombol “Tambah”. Untuk menambah data satuan, tersedia *form* untuk mengisi nama satuan. Tersedia tombol “Back” untuk balik ke halaman *master data* dan tombol “Submit” untuk menyimpan data ke dalam *database*. Tampilan halaman dapat dilihat pada gambar di atas.



Gambar 4.133 Tampilan Halaman Mengubah Data Satuan

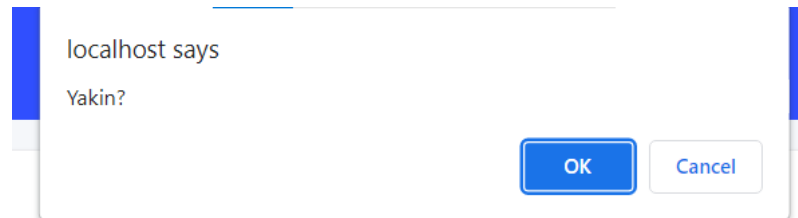
Halaman *edit data* satuan dapat diakses dengan menekan tombol “Edit”. Untuk mengubah data satuan, tersedia *form* untuk mengubah nama satuan. Tersedia tombol “Back” untuk balik ke halaman *master data* dan tombol “Submit” untuk menyimpan data ke dalam *database*. Tampilan halaman dapat dilihat pada gambar di atas.



Gambar 4.134 Tampilan notifikasi data sudah ada pada halaman *master data* satuan

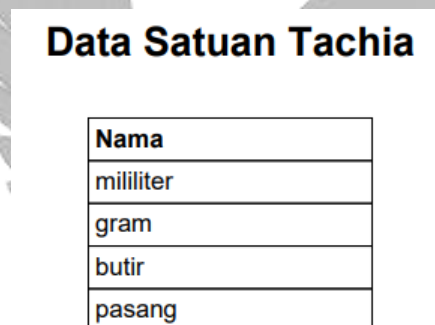
Setelah menekan tombol “*Submit*” setelah menambah dan mengubah data, jika data sudah tercantum di dalam *database*, maka sistem akan menampilkan notifikasi yang ditunjukkan pada gambar di atas.

Jika data yang berhasil tersimpan ke dalam *database*, sistem akan menampilkan halaman *Master Data User*.



Gambar 4.135 Tampilan notifikasi konfirmasi hapus data satuan

Setelah menekan tombol “*Deactivate*”, sistem akan menampilkan notifikasi konfirmasi hapus data. Jika ingin menghapus data, maka tekan tombol “OK”.



Nama
mililiter
gram
butir
pasang

Gambar 4.136 Tampilan PDF *Master Data Satuan*

Halaman PDF dapat diakses dengan menekan tombol “PDF”. Tampilan halaman dapat dilihat pada gambar di atas.

10) Halaman *Master Data* Pesanan

Data Pesanan

PDF Sort

Tanggal Awal
12/13/2022

Tanggal Akhir
12/13/2022

Show 10 entries Search:

Tanggal	Pembayaran	Kembalian	Pemasukan	Urutan	Meja	Pelanggan	Metode Bayar	Diskon	Kasir	Status	Aksi
29/11/2022 - 09:06:45	Rp.60.500	Rp.0	Rp.60.500	1	1	guest	kredit	Tidak Ada	jane doe	sudah bayar	Detail
30/11/2022 - 10:46:00	Rp.50.000	Rp.17.000	Rp.33.000	1	1	guest	kas	Tidak Ada	jane doe	sudah bayar	Detail
30/11/2022 - 11:17:14	Rp.44.000	Rp.0	Rp.44.000	2	1	steven	debit	Tidak Ada	doe doe	sudah bayar	Detail

Gambar 4.137 Tampilan Halaman *Master Data* Pesanan

Halaman *Master Data* Pesanan adalah halaman yang menampilkan *list* data setiap pesanan yang dapat menggunakan sistem usulan dalam bentuk tabel. Halaman hanya dapat diakses oleh pihak manajer. Data yang ditampilkan adalah tanggal pesanan, pembayaran, kembalian, pemasukan, urutan pesanan, nomor meja, nama pelanggan, metode pembayaran, ada atau tidaknya diskon, nama pihak kasir, dan status pesanan. Isi status pesanan terdiri dari “belum bayar” dan “sudah bayar”.

Pada bagian atas halaman telah tersedia dua tombol untuk mengakses halaman tambah data dan *sorting* data. Pada bagian kanan tabel terdapat kolom aksi yang menyediakan tombol untuk mengakses halaman *detail* data pesanan.

Pada bagian atas halaman juga terdapat kolom yang digunakan sebagai *sorting* data. Secara *default* kolom akan terisi dengan tanggal hari ini.

Tampilan halaman dapat dilihat pada gambar di atas.

Data Pesanan Tachia

Tanggal	Metode Bayar	Diskon	Pembayaran	Kembalian	Pemasukan
29/11/2022 - 09:06:45	kredit	Tidak Ada	Rp.60.500	Rp.0	Rp.60.500
30/11/2022 - 10:46:00	kas	Tidak Ada	Rp.50.000	Rp.17.000	Rp.33.000
30/11/2022 - 03:23:50	kas	Tidak Ada	Rp.20.000	Rp.9.000	Rp.11.000
05/12/2022 - 10:47:47	debit	Tidak Ada	Rp.11.000	Rp.0	Rp.11.000
09/12/2022 - 12:40:59	kredit	Tidak Ada	Rp.33.000	Rp.0	Rp.33.000
10/12/2022 - 09:01:42	kredit	Tidak Ada	Rp.22.000	Rp.0	Rp.22.000
30/11/2022 - 11:17:14	debit	Tidak Ada	Rp.44.000	Rp.0	Rp.44.000
30/11/2022 - 11:17:16	kas	Tidak Ada	Rp.77.000	Rp.44.000	Rp.33.000
12/12/2022 - 12:55:19	debit	Ada	Rp.77.000	Rp.0	Rp.77.000
Total Pemasukan					Rp.324.500

Gambar 4.138 Tampilan PDF *Master Data* Pesanan

Halaman PDF dapat diakses dengan menekan tombol “PDF”. Tampilan halaman dapat dilihat pada gambar di atas.

11) Halaman *Master Data* Pesanan *Detail*

Menu	Jumlah	Subtotal	Diskon(%)	Subtotal (Diskon)	Subtotal (Tax)
teh tawar	2	Rp.20.000	0	Rp.20.000	Rp.22.000
sate babi	1	Rp.50.000	50	Rp.25.000	Rp.27.500

Gambar 4.139 Tampilan Halaman *Master Data* Pesanan *Detail*

Halaman *Master Data* Pesanan *Detail* adalah halaman yang menampilkan *list detail* data setiap pesanan. Halaman hanya dapat diakses oleh pihak manajer dengan menekan tombol “*Detail*” pada halaman *Master Data* Pesanan. Data yang ditampilkan adalah nama *menu*, jumlah *menu*, subtotal, persentase diskon, subtotal diskon, dan subtotal pajak. Pada bagian atas halaman ditampilkan urutan pesanan, nomor meja, dan waktu pesanan terbuat. Tampilan halaman dapat dilihat pada gambar di atas.

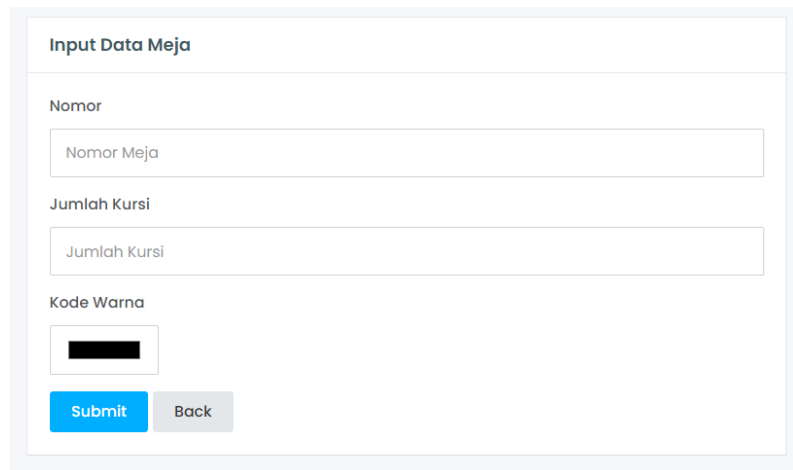
12) Halaman *Master Data* Meja

Nomor Meja	Jumlah Kursi	Kode Warna	Status Tamu	Aksi
1	4	#C8FFD4	kosong	Edit Deactivate
2	4	#BBE8FC	kosong	Edit Deactivate
3	4	#B1AFFE	kosong	Edit Deactivate
4	4	#FDFD8D	kosong	Edit Deactivate
5	2	#BCCEF8	kosong	Edit Deactivate

Gambar 4.140 Tampilan Halaman *Master Data* Meja

Halaman *Master Data* Meja adalah halaman yang menampilkan *list* data setiap meja dalam bentuk tabel. Halaman hanya dapat diakses oleh pihak manajer. Data yang ditampilkan adalah nomor meja, jumlah kursi, kode warna, dan status kehadiran tamu.

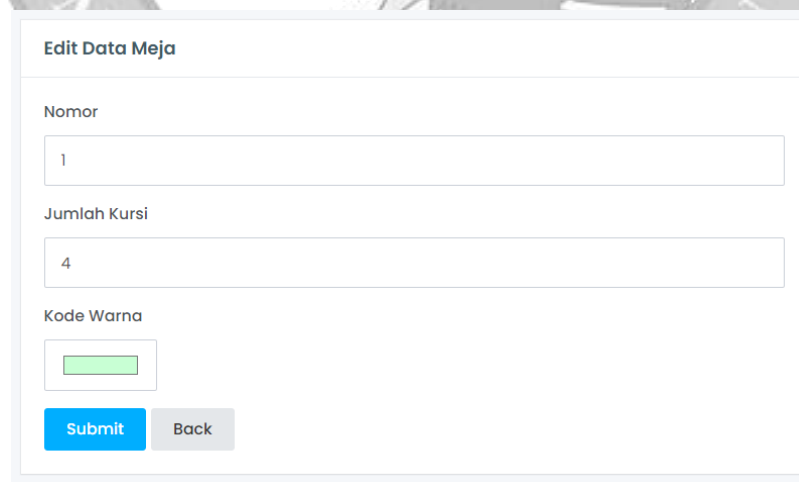
Pada bagian atas halaman telah tersedia dua tombol untuk mengakses halaman tambah data dan halaman PDF. Pada bagian kanan tabel terdapat kolom aksi yang menyediakan tombol untuk mengakses halaman *edit* data dan *deactivate* data. Tampilan halaman dapat dilihat pada gambar di atas.



The screenshot shows a form titled "Input Data Meja". It has three input fields: "Nomor" (empty), "Jumlah Kursi" (empty), and "Kode Warna" (black). At the bottom, there are two buttons: "Submit" (blue) and "Back" (grey).

Gambar 4.141 Tampilan Halaman Menambah Data Meja

Halaman *input data* meja dapat diakses dengan menekan tombol “Tambah”. Untuk menambah data meja, tersedia *form* untuk mengisi nomor meja, jumlah kursi, dan kode warna. Tersedia tombol “Back” untuk balik ke halaman *master data* dan tombol “Submit” untuk menyimpan data ke dalam *database*. Tampilan halaman dapat dilihat pada gambar di atas.

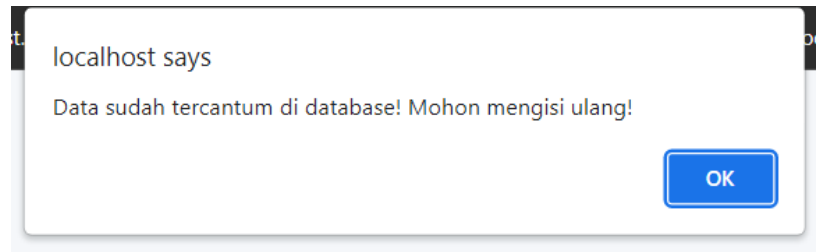


The screenshot shows a form titled "Edit Data Meja". It has three input fields: "Nomor" (1), "Jumlah Kursi" (4), and "Kode Warna" (light green). At the bottom, there are two buttons: "Submit" (blue) and "Back" (grey).

Gambar 4.142 Tampilan Halaman Mengubah Data Meja

Halaman *edit data* meja dapat diakses dengan menekan tombol “Edit”. Untuk mengubah data meja, tersedia *form* untuk mengubah nomor meja, jumlah kursi, dan kode warna. Tersedia tombol “Back” untuk balik ke halaman *master data* dan tombol

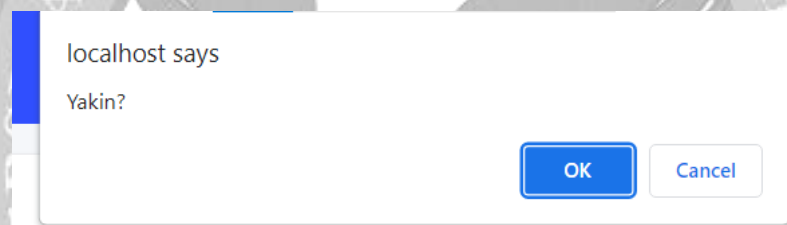
“*Submit*” untuk menyimpan data ke dalam *database*. Tampilan halaman dapat dilihat pada gambar di atas.



Gambar 4.143 Tampilan notifikasi data sudah ada pada halaman *master data* meja

Setelah menekan tombol “*Submit*” setelah menambah dan mengubah data, jika data sudah tercantum di dalam *database*, maka sistem akan menampilkan notifikasi yang ditunjukkan pada gambar di atas.

Jika data yang berhasil tersimpan ke dalam *database*, sistem akan menampilkan halaman *Master Data Meja*.



Gambar 4.144 Tampilan notifikasi konfirmasi hapus data meja

Setelah menekan tombol “*Deactivate*”, sistem akan menampilkan notifikasi konfirmasi hapus data. Jika ingin menghapus data, maka tekan tombol “OK”.

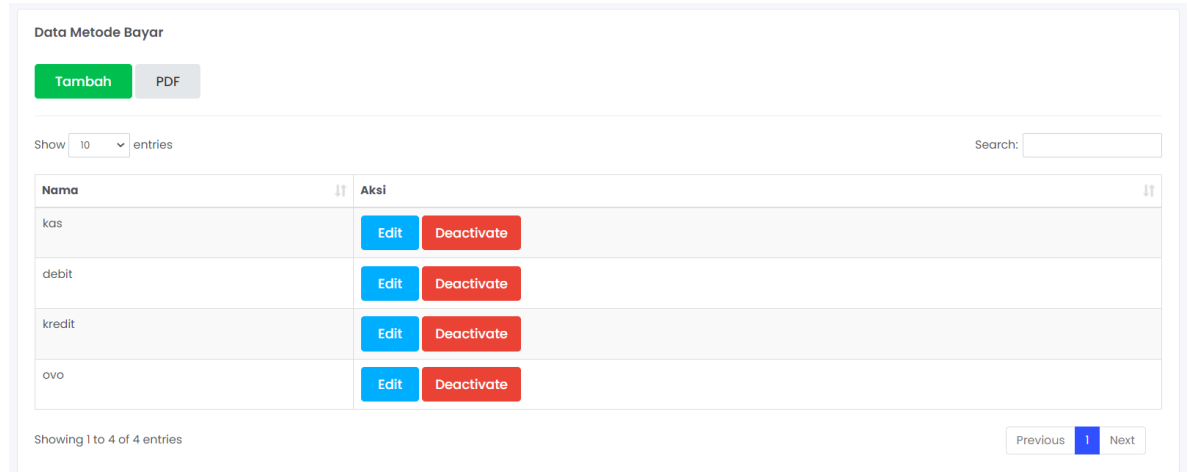
Data Meja Tachia

Nomor Meja	Jumlah Kursi
1	4
2	4
3	4
4	4
5	2
6	2
7	1

Gambar 4.145 Tampilan PDF *Master Data Meja*

Halaman PDF dapat diakses dengan menekan tombol “PDF”. Tampilan halaman dapat dilihat pada gambar di atas.

13) Halaman *Master Data* Metode Bayar



Gambar 4.146 Tampilan Halaman *Master Data* Metode Bayar

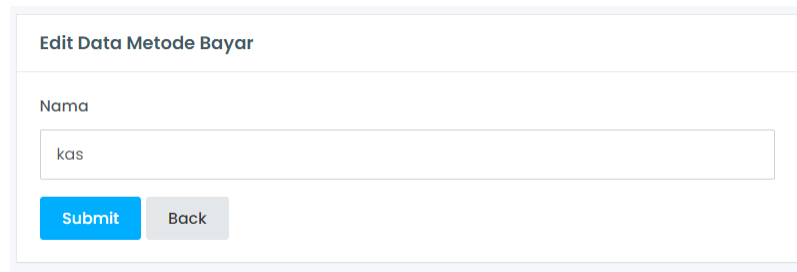
Halaman *Master Data* Metode Bayar adalah halaman yang menampilkan *list* data setiap metode bayar dalam bentuk tabel. Halaman hanya dapat diakses oleh pihak manajer. Data yang ditampilkan adalah nama metode bayar.

Pada bagian atas halaman telah tersedia dua tombol untuk mengakses halaman tambah data dan halaman PDF. Pada bagian kanan tabel terdapat kolom aksi yang menyediakan tombol untuk mengakses halaman *edit* data dan *deactivate* data. Tampilan halaman dapat dilihat pada gambar di atas.

Nama
Nama Metode Bayar

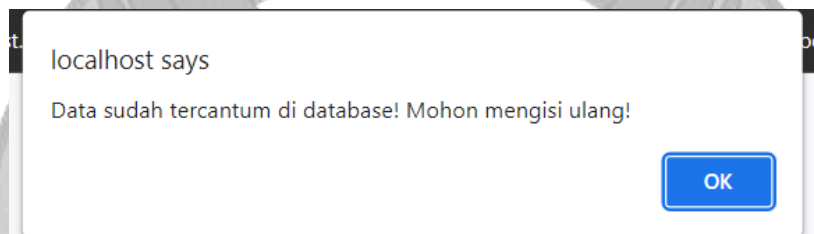
Gambar 4.147 Tampilan Halaman Menambah Data Metode Bayar

Halaman *input data* metode bayar dapat diakses dengan menekan tombol “Tambah”. Untuk menambah data metode bayar, tersedia *form* untuk mengisi nama metode bayar. Tersedia tombol “Back” untuk balik ke halaman *master data* dan tombol “Submit” untuk menyimpan data ke dalam *database*. Tampilan halaman dapat dilihat pada gambar di atas.



Gambar 4.148 Tampilan Halaman Mengubah Data Metode Bayar

Halaman *edit data* metode bayar dapat diakses dengan menekan tombol “*Edit*”. Untuk mengubah data metode bayar, tersedia *form* untuk mengubah nama metode bayar. Tersedia tombol “*Back*” untuk balik ke halaman *master data* dan tombol “*Submit*” untuk menyimpan data ke dalam *database*. Tampilan halaman dapat dilihat pada gambar di atas.



Gambar 4.149 Tampilan notifikasi data sudah ada pada halaman *master data* metode bayar

Setelah menekan tombol “*Submit*” setelah menambah dan mengubah data, jika data sudah tercantum di dalam *database*, maka sistem akan menampilkan notifikasi yang ditunjukkan pada gambar di atas.

Jika data yang berhasil tersimpan ke dalam *database*, sistem akan menampilkan halaman *Master Data Metode Bayar*.



Gambar 4.150 Tampilan notifikasi konfirmasi hapus data metode bayar

Setelah menekan tombol “*Deactivate*”, sistem akan menampilkan notifikasi konfirmasi hapus data. Jika ingin menghapus data, maka tekan tombol “*OK*”.

Data Metode Bayar Tachia

Nama
kas
debit
kredit
ovo

Gambar 4.151 Tampilan PDF *Master Data* Metode Bayar

Halaman PDF dapat diakses dengan menekan tombol “PDF”. Tampilan halaman dapat dilihat pada gambar di atas.

14) Halaman *Master Data* Pelanggan

Nama	No Telp	Status Hadir	Aksi
guest	080000000000	belum hadir	Edit Deactivate
steven	087890123456	belum hadir	Edit Deactivate
john doe	080000000001	belum hadir	Edit Deactivate

Gambar 4.152 Tampilan Halaman *Master Data* Pelanggan

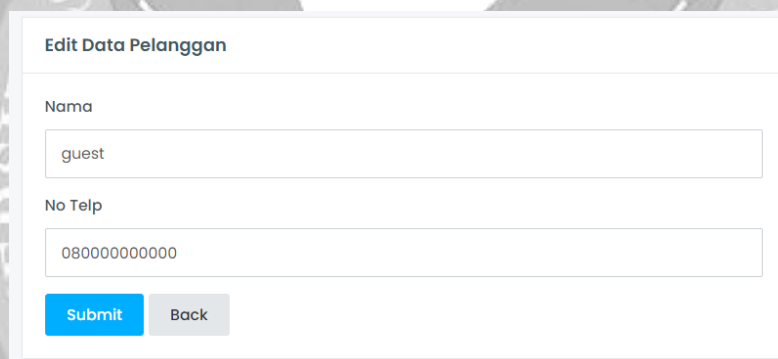
Halaman *Master Data* Pelanggan adalah halaman yang menampilkan *list* data setiap pelanggan usulan dalam bentuk tabel. Halaman hanya dapat diakses oleh pihak manajer. Data yang ditampilkan adalah nama pelanggan, nomor telepon pelanggan, dan status kehadiran.

Pada bagian atas halaman telah tersedia dua tombol untuk mengakses halaman tambah data dan halaman PDF. Pada bagian kanan tabel terdapat kolom aksi yang menyediakan tombol untuk mengakses halaman *edit* data dan *deactivate* data. Tampilan halaman dapat dilihat pada gambar di atas.



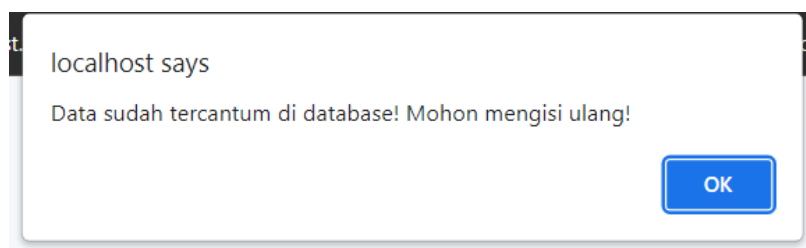
Gambar 4.153 Tampilan Halaman Menambah Data Pelanggan

Halaman *input data* pelanggan dapat diakses dengan menekan tombol “Tambah”. Untuk menambah data pelanggan, tersedia *form* untuk mengisi nama pelanggan dan nomor telepon pelanggan. Tersedia tombol “Back” untuk balik ke halaman *master data* dan tombol “Submit” untuk menyimpan data ke dalam *database*. Tampilan halaman dapat dilihat pada gambar di atas.



Gambar 4.154 Tampilan Halaman Mengubah Data Pelanggan

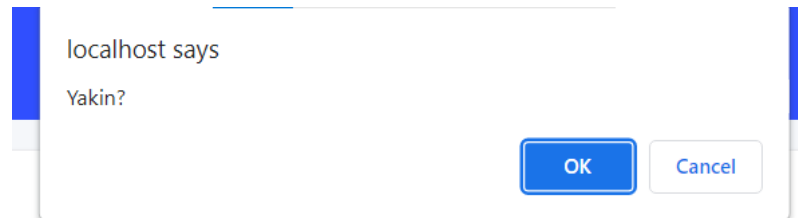
Halaman *edit data* pelanggan dapat diakses dengan menekan tombol “Edit”. Untuk mengubah data pelanggan, tersedia *form* untuk mengubah nama pelanggan, dan nomor telepon pelanggan. Tersedia tombol “Back” untuk balik ke halaman *master data* dan tombol “Submit” untuk menyimpan data ke dalam *database*. Tampilan halaman dapat dilihat pada gambar di atas.



Gambar 4.155 Tampilan notifikasi data sudah ada pada halaman *master data* pelanggan

Setelah menekan tombol “*Submit*” setelah menambah dan mengubah data, jika data sudah tercantum di dalam *database*, maka sistem akan menampilkan notifikasi yang ditunjukkan pada gambar di atas.

Jika data yang berhasil tersimpan ke dalam *database*, sistem akan menampilkan halaman *Master Data* Pelanggan.



Gambar 4.156 Tampilan notifikasi konfirmasi hapus data pelanggan

Setelah menekan tombol “*Deactivate*”, sistem akan menampilkan notifikasi konfirmasi hapus data. Jika ingin menghapus data, maka tekan tombol “OK”.

Data Pelanggan Tachia		
Nama	Nomor Telepon	Status Hadir
guest	080000000000	belum hadir
steven	087890123456	belum hadir
john doe	080000000001	belum hadir

Gambar 4.157 Tampilan PDF *Master Data* Pelanggan

Halaman PDF dapat diakses dengan menekan tombol “PDF”. Tampilan halaman dapat dilihat pada gambar di atas.

15) Halaman *Master Data* Pengeluaran

Data Pengeluaran

Tambah PDF Sort

Tanggal Awal
12/13/2022

Tanggal Akhir
12/13/2022

Show 10 entries Search:

Nama User	Date	Total Pengeluaran	Keterangan	Kategori	Supplier	Aksi
john doe	03/11/2022	Rp.50.000	air	bahan makan	superindo	Edit
john doe	28/11/2022	Rp.100.000	sumpit	alat makan	superindo	Edit
john doe	20/10/2022	Rp.250.000	bumbu bumbu	bahan makan	superindo	Edit
john doe	12/12/2022	Rp.50.000	air	bahan makan	superindo	Edit

Gambar 4.158 Tampilan Halaman *Master Data* Pengeluaran

Halaman *Master Data* Pengeluaran adalah halaman yang menampilkan *list* data setiap pengeluaran dalam bentuk tabel. Halaman hanya dapat diakses oleh pihak manajer. Data yang ditampilkan adalah nama *user*, tanggal pengeluaran, total harga pengeluaran, keterangan pengeluaran, kategori pengeluaran, dan nama *supplier*.

Pada bagian atas halaman telah tersedia tiga tombol untuk mengakses halaman tambah data, halaman PDF, dan *sorting* data. Pada bagian kanan tabel terdapat kolom aksi yang menyediakan tombol untuk mengakses halaman *edit* data dan *deactivate* data.

Pada bagian atas halaman juga terdapat kolom yang digunakan sebagai *sorting* data. Secara *default* kolom akan terisi dengan tanggal hari ini. Tampilan halaman dapat dilihat pada gambar di atas.

Input Data Pengeluaran

Harga

Keterangan

Kategori

Supplier

Tanggal

Gambar 4.159 Tampilan Halaman Menambah Data Pengeluaran

Halaman *input data* pengeluaran dapat diakses dengan menekan tombol “Tambah”. Untuk menambah data pengeluaran, tersedia *form* untuk mengisi harga pengeluaran, keterangan pengeluaran, kategori pengeluaran, nama *supplier*, dan tanggal pengeluaran. Tersedia tombol “Back” untuk balik ke halaman *master data* dan tombol “Submit” untuk menyimpan data ke dalam *database*. Tampilan halaman dapat dilihat pada gambar di atas.

Edit Data Pengeluaran

Harga

Keterangan

Kategori

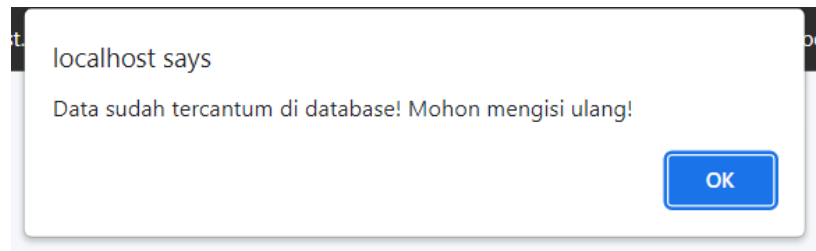
Supplier

Tanggal

Gambar 4.160 Tampilan Halaman Mengubah Data Pengeluaran

Halaman *edit data* pengeluaran dapat diakses dengan menekan tombol “Edit”. Untuk mengubah data pengeluaran, tersedia *form* untuk mengubah na harga

pengeluaran, keterangan pengeluaran, kategori pengeluaran, nama *supplier*, dan tanggal pengeluaran. Tersedia tombol “Back” untuk balik ke halaman *master data* dan tombol “Submit” untuk menyimpan data ke dalam *database*. Tampilan halaman dapat dilihat pada gambar di atas.



Gambar 4.161 Tampilan notifikasi data sudah ada pada halaman *master data* pengeluaran

Setelah menekan tombol “Submit” setelah menambah dan mengubah data, jika data sudah tercantum di dalam *database*, maka sistem akan menampilkan notifikasi yang ditunjukkan pada gambar di atas.

Jika data yang berhasil tersimpan ke dalam *database*, sistem akan menampilkan halaman *Master Data* Pengeluaran.

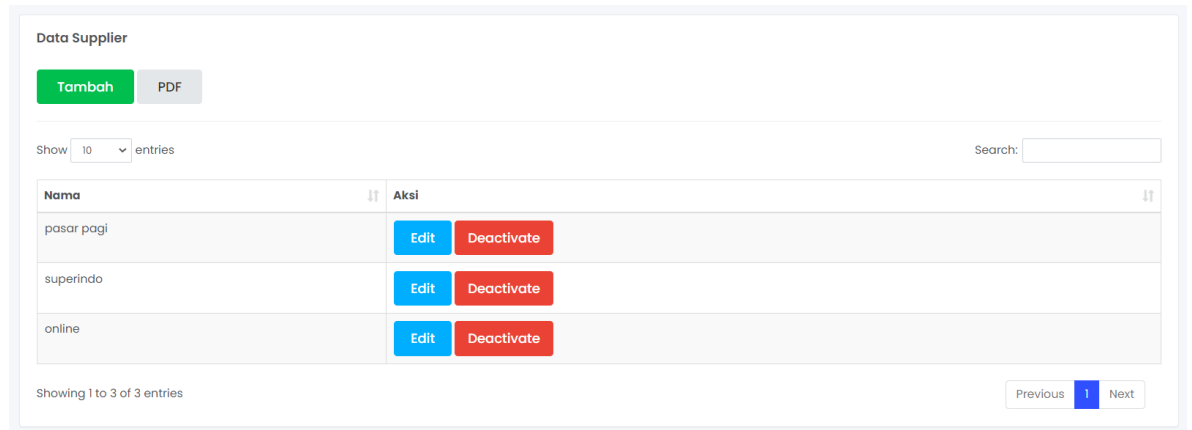
Data Pengeluaran Tachia

Tanggal	Supplier	Kategori	Keterangan	Total
03/11/2022	superindo	bahan makan	air	Rp.50.000
28/11/2022	superindo	alat makan	sumpit	Rp.100.000
20/10/2022	superindo	bahan makan	bumbu bumbu	Rp.250.000
12/12/2022	superindo	bahan makan	air	Rp.50.000
Total Pengeluaran				Rp.450.000

Gambar 4.162 Tampilan PDF *Master Data* Pengeluaran

Halaman PDF dapat diakses dengan menekan tombol “PDF”. Tampilan halaman dapat dilihat pada gambar di atas.

16) Halaman *Master Data Supplier*



Gambar 4.163 Tampilan Halaman *Master Data Supplier*

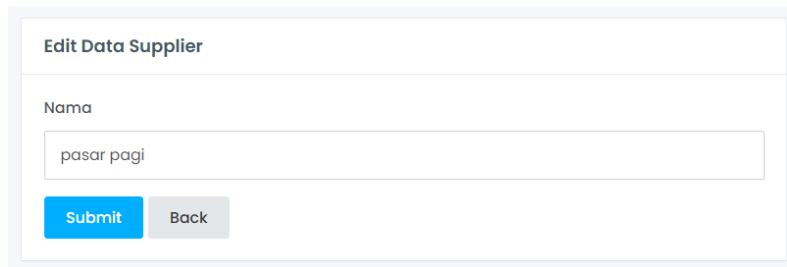
Halaman *Master Data Supplier* adalah halaman yang menampilkan *list* data setiap *supplier* yang dapat menggunakan sistem usulan dalam bentuk tabel. Halaman hanya dapat diakses oleh pihak manajer. Data yang ditampilkan adalah nama *supplier*.

Pada bagian atas halaman telah tersedia dua tombol untuk mengakses halaman tambah data dan halaman PDF. Pada bagian kanan tabel terdapat kolom aksi yang menyediakan tombol untuk mengakses halaman *edit* data dan *deactivate* data. Tampilan halaman dapat dilihat pada gambar di atas.

The screenshot shows a form titled "Input Data Supplier". It contains a single text input field with the label "Nama" and the placeholder text "Nama Supplier". Below the input field, there are two buttons: "Submit" (blue) and "Back" (grey).

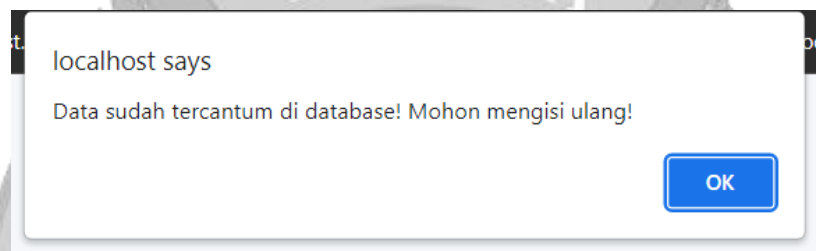
Gambar 4.164 Tampilan Halaman Menambah Data *Supplier*

Halaman *input data user* dapat diakses dengan menekan tombol "Tambah". Untuk menambah data *user*, tersedia *form* untuk mengisi nama *supplier*. Tersedia tombol "Back" untuk balik ke halaman *master data* dan tombol "Submit" untuk menyimpan data ke dalam *database*. Tampilan halaman dapat dilihat pada gambar di atas.



Gambar 4.165 Tampilan Halaman Mengubah Data *Supplier*

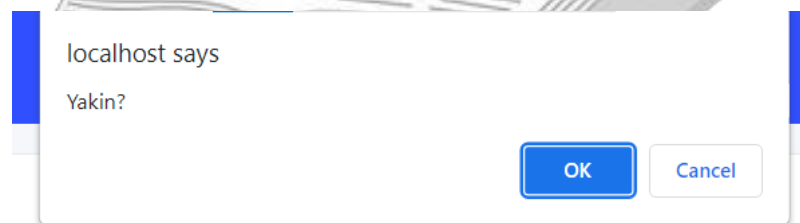
Halaman *edit data user* dapat diakses dengan menekan tombol “*Edit*”. Untuk mengubah data *user*, tersedia *form* untuk mengubah nama *supplier*. Tersedia tombol “*Back*” untuk balik ke halaman *master data* dan tombol “*Submit*” untuk menyimpan data ke dalam *database*. Tampilan halaman dapat dilihat pada gambar di atas.



Gambar 4.166 Tampilan notifikasi data sudah ada pada halaman *master data supplier*

Setelah menekan tombol “*Submit*” setelah menambah dan mengubah data, jika data sudah tercantum di dalam *database*, maka sistem akan menampilkan notifikasi yang ditunjukkan pada gambar di atas.

Jika data yang berhasil tersimpan ke dalam *database*, sistem akan menampilkan halaman *Master Data Supplier*.



Gambar 4.167 Tampilan notifikasi konfirmasi hapus data *supplier*

Setelah menekan tombol “*Deactivate*”, sistem akan menampilkan notifikasi konfirmasi hapus data. Jika ingin menghapus data, maka tekan tombol “*OK*”.

Data Supplier Tachia

Nama
pasar pagi
superindo
online

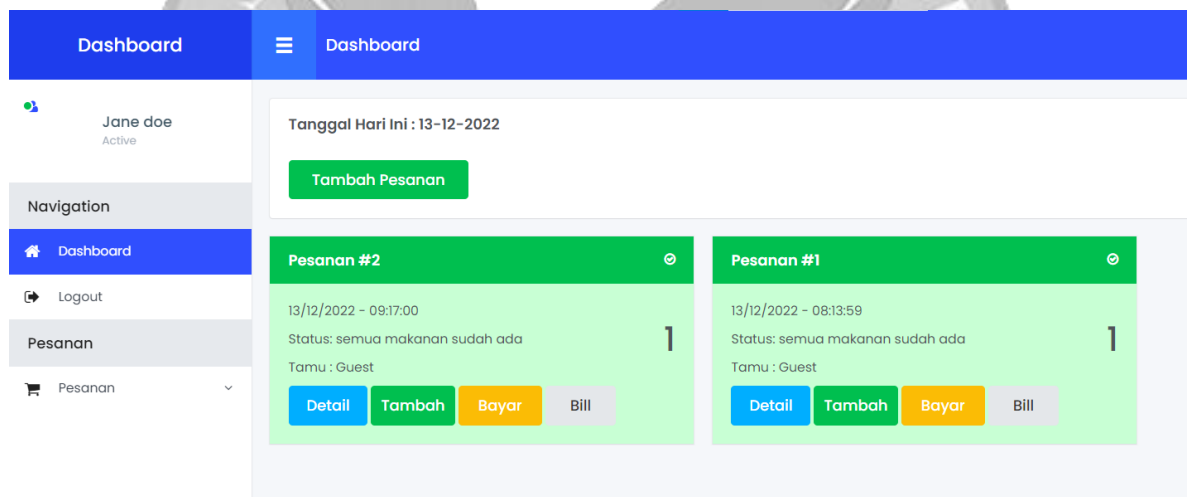
Gambar 4.168 Tampilan PDF *Master Data Supplier*

Halaman PDF dapat diakses dengan menekan tombol “PDF”. Tampilan halaman dapat dilihat pada gambar di atas.

4.4.2.3. Perancangan Lapisan Antarmuka (Pihak Kasir)

Berikut merupakan tampilan halaman yang dapat diakses oleh pihak kasir.

1) Halaman *Dashboard* Kasir



Gambar 4.169 Tampilan Halaman *Dashboard* Kasir



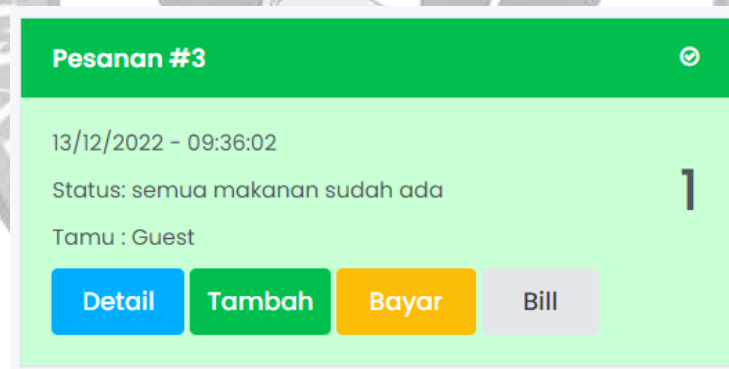
Gambar 4.170 Tampilan *Card* Pesanan Status Makanan Belum Dianter



Gambar 4.171 Tampilan *Card* Pesanan Status Beberapa Makanan Sudah Ada



Gambar 4.172 Tampilan *Card* Pesanan Status Semua Makanan Sudah Ada



Gambar 4.173 Tampilan *Card* Pesanan Status Selesai

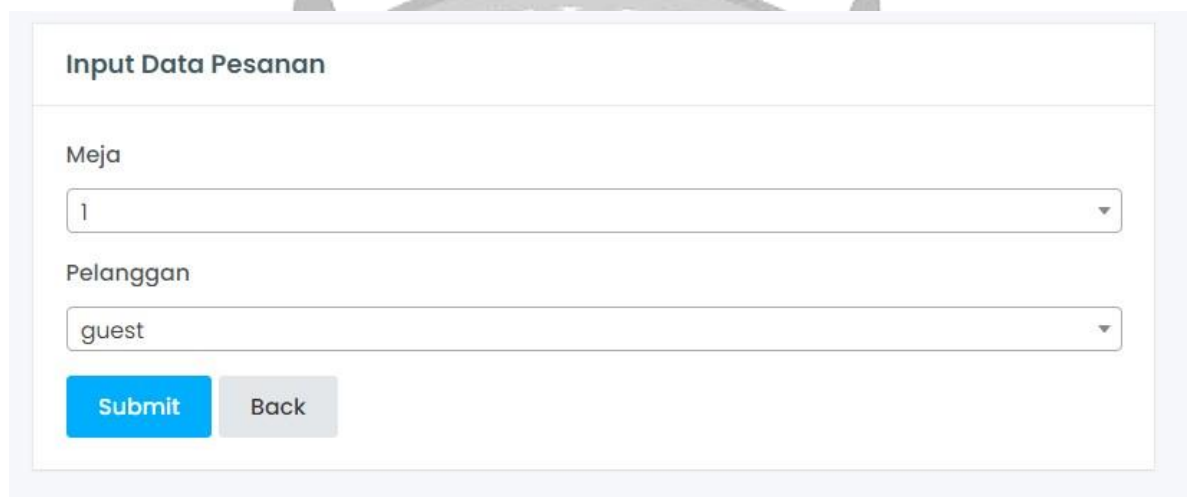
Pada halaman *dashboard* kasir tersedia pesanan-pesanan yang dibuat pada hari itu. Pesanan ditampilkan dalam bentuk *card*. Setiap warna *background* pada *card* akan berubah tergantung dengan nomor meja yang dipilih. Pada bagian atas halaman tersedia tanggal pada hari ini dan tombol untuk menambah pesanan.

Setiap *card* pesanan tersedia nomor urutan pesanan pada hari ini, waktu pesanan terbuat, status pesanan, tamu, dan nomor meja pada sebelah kanan. Tersedia tombol untuk mengelola *detail* pesanan dan menambah data *detail* pesanan. Status pesanan

pada *card* terdiri dari “makanan belum dianter”, “beberapa makanan sudah ada” dan “semua makanan sudah ada”.

Bagian *header card* memiliki warna yang berhubungan dengan status pesanan. Jika status pesanan makanan belum ada, maka warnanya menjadi abu-abu dan hanya tersedia tombol “*Detail*” dan “*Tambah*”. Jika status pesanan beberapa makanan sudah ada, maka warnanya menjadi kuning. Jika status pesanan semua makanan sudah ada, maka warnanya menjadi hijau dan tersedia tambahan tombol “*Bayar*”. Jika sudah melakukan pembayaran, maka tersedia tombol “*Bill*”.

2) Mengelola Pesanan Kasir



Input Data Pesanan

Meja
1

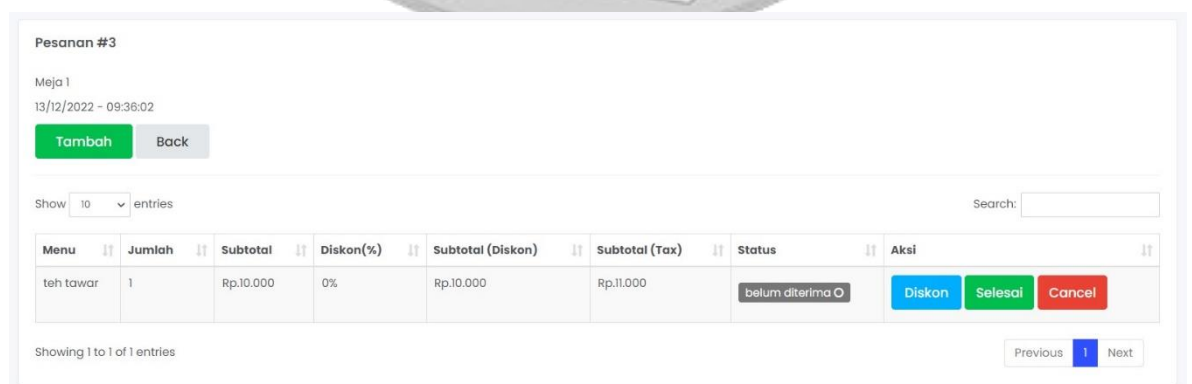
Pelanggan
guest

Submit Back

Gambar 4.174 Tampilan Halaman Menambah Data *Pesanan*

Pada halaman *input data pesanan*, tersedia *form* untuk mengisi nomor meja dan nama pelanggan. Pada bagian bawah tersedia tombol “*Back*” untuk balik ke halaman *dashboard*, dan tombol “*Submit*” untuk menyimpan data pesanan.

3) Mengelola Pesanan *Detail*



Pesanan #3

Meja 1
13/12/2022 - 09:38:02

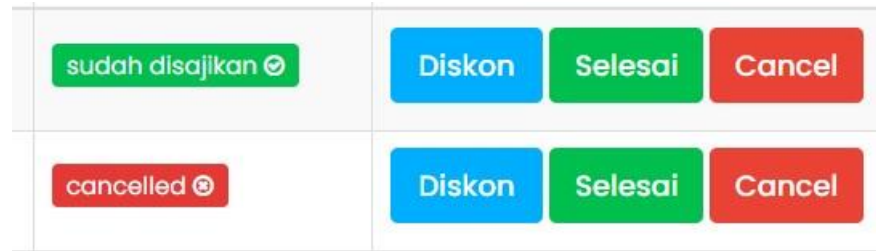
Tambah Back

Show 10 entries Search:

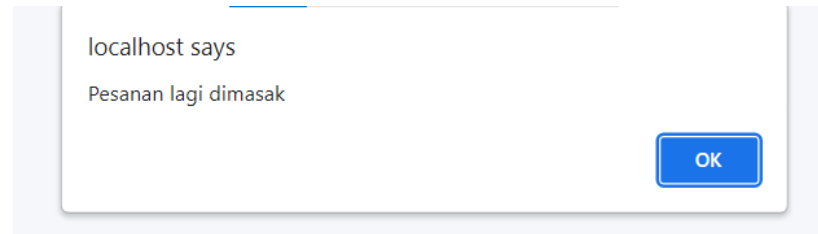
Menu	Jumlah	Subtotal	Diskon(%)	Subtotal (Diskon)	Subtotal (Tax)	Status	Aksi
teh tawar	1	Rp.10.000	0%	Rp.10.000	Rp.11.000	belum diterima	Diskon Selesai Cancel

Showing 1 to 1 of 1 entries Previous 1 Next

Gambar 4.175 Tampilan Halaman Mengelola *Detail* Pesanan



Gambar 4. 176 Tampilan Status *Detail* Pesanan



Gambar 4.177 Tampilan notifikasi pesanan lagi dimasak pada halaman *detail* pesanan

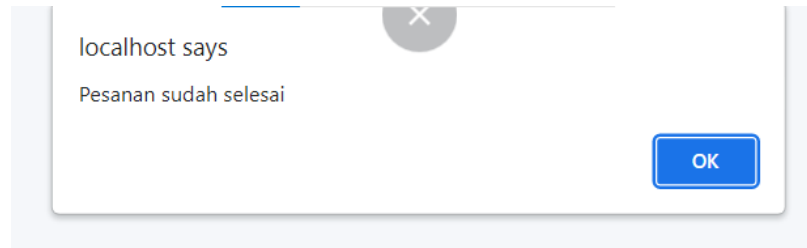
Halaman pesanan *detail* dapat diakses melalui halaman *dashboard* kasir setelah menekan tombol “*Detail*”. Pada bagian atas halaman ditampilkan nomor urutan pesanan, nomor meja, dan waktu pesanan. Tersedia juga dua tombol untuk menambah data pesanan *detail* dan balik ke halaman *dashboard*.

Pada halaman pesanan *detail* ditampilkan data dalam bentuk tabel. Yang ditampilkan adalah nama *menu*, jumlah *menu*, subtotal, persentase diskon, *subtotal* diskon, *subtotal tax*, dan status penyajian makanan. Pada sebelah kanan tabel tersedia button untuk memberikan diskon, menyelesaikan pesanan, dan membatalkan pesanan. Isi status penyajian terdiri dari “belum diterima”, “lagi dimasak”, “sudah selesai”, dan “sudah disajikan”.

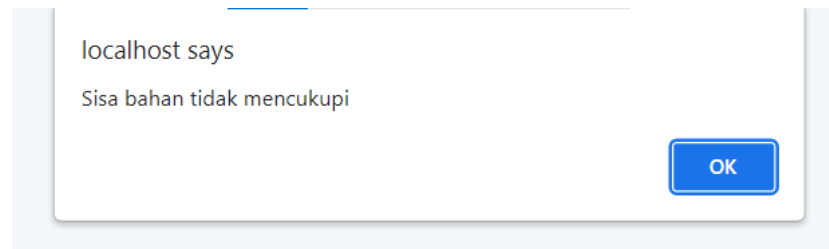
Ketika menekan tombol “Selesai”, status penyajian makanan akan berubah dari “sudah selesai” menjadi “sudah disajikan”. Jika status pesanan masih “lagi dimasak”, maka sistem akan menampilkan notifikasi yang ditunjukkan pada gambar di atas.

The image shows a form titled 'Input Menu Pesanan #3'. It has two main sections. The first section is labeled 'Menu' and contains a dropdown menu with 'teh tawar' selected. The second section is labeled 'Jumlah' and contains a text input field with the number '1'. At the bottom of the form are two buttons: 'Submit' (blue) and 'Back' (grey).

Gambar 4.178 Tampilan Halaman *Input Data Detail Pesanan*



Gambar 4.179 Tampilan notifikasi pesanan sudah selesai pada halaman *detail* pesanan

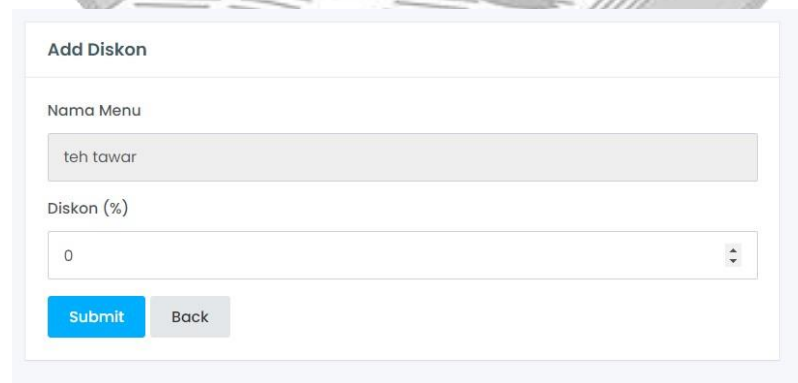


Gambar 4.180 Tampilan notifikasi bahan tidak cukup pada halaman *detail* pesanan

Halaman *input* pesanan *detail* dapat diakses melalui halaman *dashboard* atau pesanan *detail* dengan menekan tombol “Tambah”. Pada halaman ini tersedia *form* untuk mengisi nama *menu* dan jumlah *menu*. Pada bagian bawah tersedia tombol “Back” untuk balik ke halaman *detail* pesanan, dan tombol “Submi” untuk menyimpan data pesanan *detail*.

Jika pesanan sudah selesai, maka sistem akan menampilkan notifikasi pesanan sudah selesai ketika menekan tombol “Tambah”.

Jika sisa bahan tidak mencukupi, maka sistem akan menampilkan notifikasi bahan tidak mencukupi.

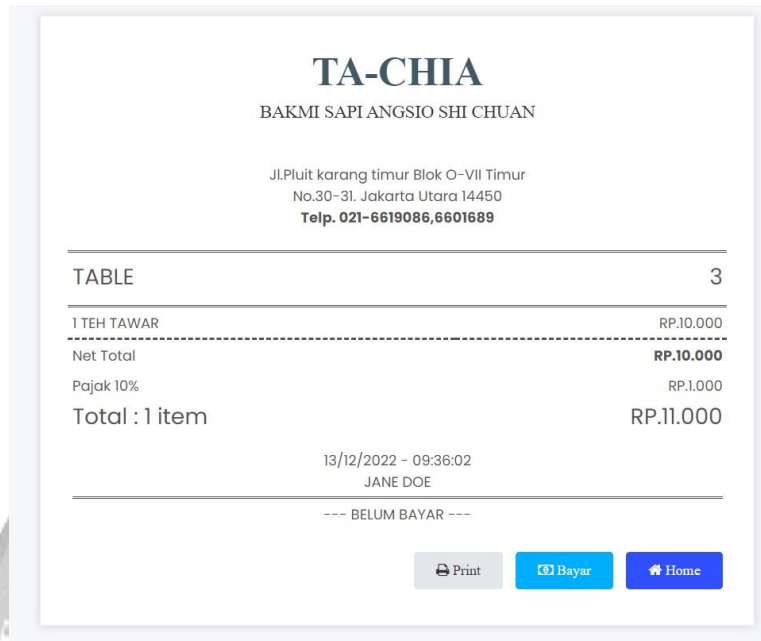


Gambar 4.181 Tampilan Halaman *Add Diskon*

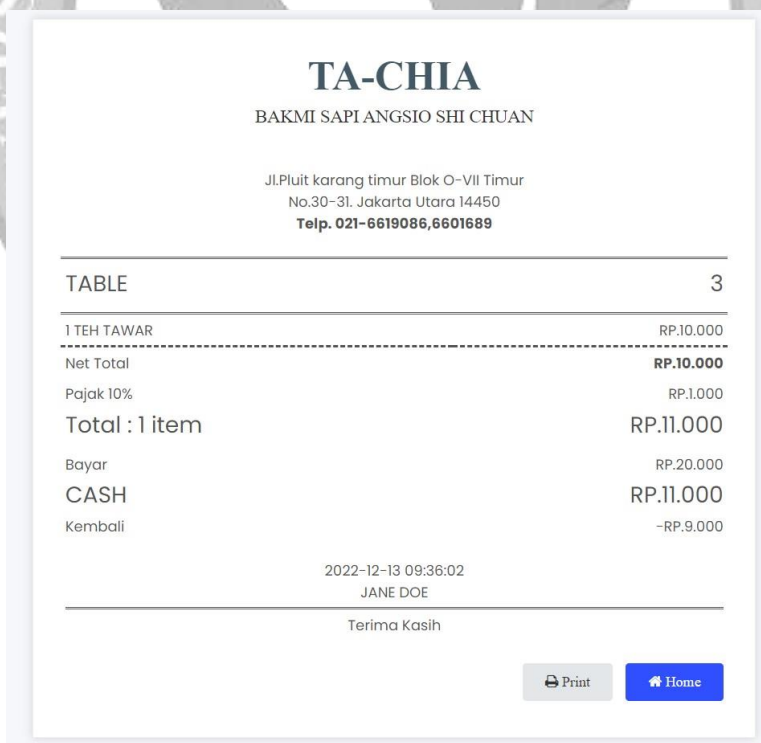
Halaman *Add Diskon* dapat diakses melalui halaman *detail* pesanan dengan menekan tombol “Diskon”. Pada halaman ini tersedia *form* untuk mengisi jumlah

diskon. Pada bagian bawah tersedia tombol “Back” untuk balik ke halaman pesanan *detail*, dan tombol “Submit” untuk menyimpan data diskon.

4) Halaman Pembayaran dan *Bill*



Gambar 4.182 Tampilan *Bill* Penagihan

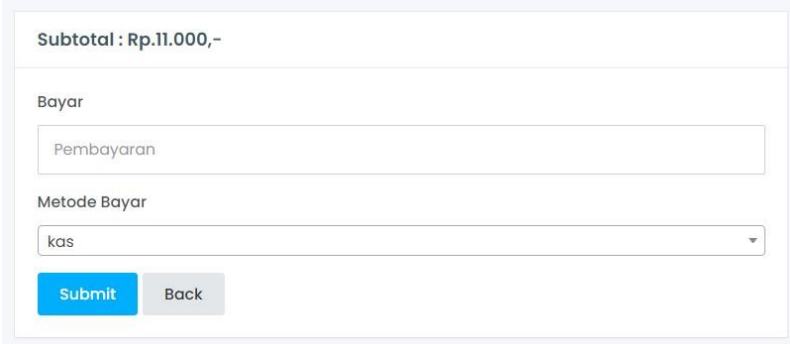


Gambar 4.183 Tampilan *Bill* Pelunasan

Bill penagihan dapat diakses melalui halaman *dashboard* dengan menekan tombol “Bayar”. Halaman ini menampilkan *bill* penagihan pesanan. Pada bagian bawah

halaman tersedia tombol untuk menampilkan halaman PDF *bill* penagihan, tombol untuk melakukan pembayaran, dan tombol untuk baik ke halaman *dashboard*.

Bill pelunasan dapat diakses setelah melakukan pembayaran. Halaman ini menampilkan *bill* pelunasan pesanan. Pada bagian bawah halaman tersedia tombol untuk menampilkan halaman PDF *bill* pelunasan dan tombol untuk balik ke halaman *dashboard*.



Subtotal : Rp.11.000,-

Bayar

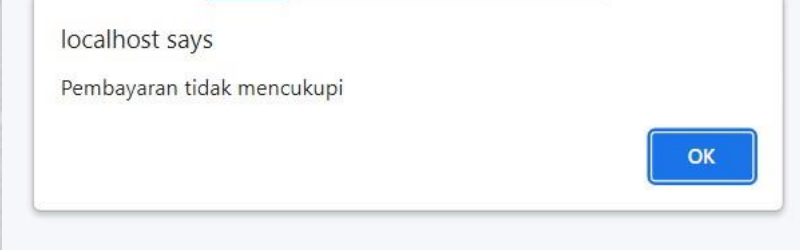
Pembayaran

Metode Bayar

kas

Submit Back

Gambar 4.184 Tampilan Halaman Pembayaran

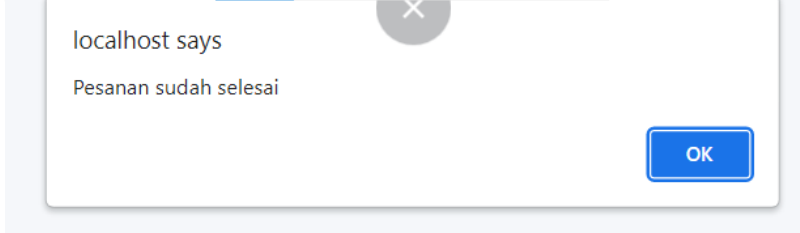


localhost says

Pembayaran tidak mencukupi

OK

Gambar 4.185 Tampilan notifikasi pembayaran tidak cukup



localhost says

Pesanan sudah selesai

OK

Gambar 4.186 Tampilan notifikasi pesanan sudah selesai pada halaman pembayaran

Halaman pembayaran dapat diakses melalui halaman *bill* penagihan dengan menekan tombol “Bayar”. Pada halaman ini tersedia *form* untuk mengisi jumlah pembayaran dan metode pembayaran. Pada bagian atas ditampilkan subtotal pesanan yang harus dibayar. Pada bagian bawah tersedia tombol “Back” untuk balik ke halaman sebelumnya, dan tombol “Submit” untuk menyimpan data pembayaran.

Jika jumlah pembayaran yang diisi kurang dari jumlah subtotal, maka sistem akan menampilkan notifikasi pembayaran tidak cukup.

Jika status pesanan sudah selesai, maka sistem akan menampilkan notifikasi pesanan sudah selesai ketika ingin membuka halaman pembayaran dari halaman *bill* penagihan.

5) Halaman Pengecekan Data Pesanan

Tanggal Hari Ini : 13-12-2022

Show 10 entries Search:

Tanggal	Pembayaran	Kembalian	Pemasukan	Urutan	Meja	Pelanggan	Metode Bayar	Kasir	Status	Aksi
2022-12-13 08:13:59	Rp.49.500	Rp.0	Rp.49.500	1	1	guest	debit	jane doe	sudah bayar	Detail Bill
2022-12-13 09:17:00	Rp.50.000	Rp.28.000	Rp.22.000	2	1	guest	kas	jane doe	sudah bayar	Detail Bill
2022-12-13 09:36:02	Rp.20.000	Rp.9.000	Rp.11.000	3	1	guest	kas	jane doe	sudah bayar	Detail Bill

Showing 1 to 3 of 3 entries Previous 1 Next

Gambar 4.187 Tampilan Halaman Pengecekan Data Pesanan

Pada halaman pengecekan data pesanan ditampilkan semua data pesanan pada hari ini dalam bentuk tabel. Pada bagian kanan tersedia tombol untuk mengakses halaman *detail* pesanan dan halaman *bill* pelunasan. Isi status pesanan terdiri dari “belum bayar” dan “sudah bayar”.

4.4.2.4. Perancangan Lapisan Antarmuka (Pihak Dapur)

Berikut merupakan tampilan halaman yang dapat diakses oleh pihak dapur.

1) Halaman *Dashboard* Dapur

Dashboard Doe doe Active

Navigation Dashboard Logout

Tanggal Hari ini : 13-12-2022

Pesanan	Meja	Menu	Jumlah	Status	Aksi
1	1	sate babi	1	sudah disajikan	Terima Selesai
1	1	teh tawar	2	sudah disajikan	Terima Selesai
2	1	kopi	1	sudah disajikan	Terima Selesai
3	1	teh tawar	1	belum diterima	Terima Selesai

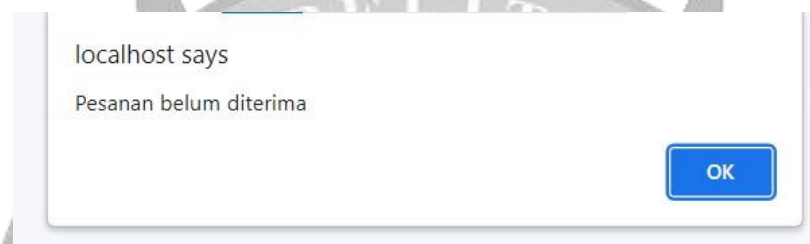
Gambar 4.188 Tampilan Halaman *Dashboard* Dapur

Pada halaman *dashboard* dapur tersedia data pesanan *detail* yang dibuat pada hari ini. Data *detail* pesanan ditampilkan dalam bentuk tabel. Yang ditampilkan adalah nomor urutan pesanan, nomor meja, nama *menu*, jumlah *menu*, dan status penyajian.

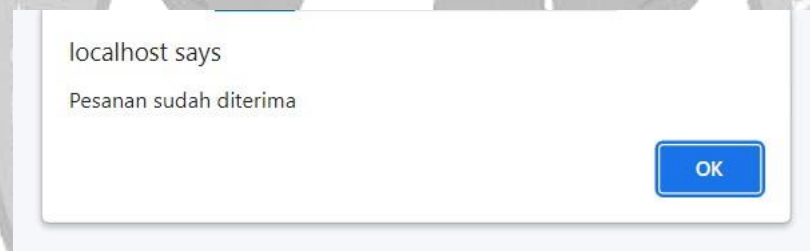
Pada bagian kanan tabel tersedia tombol untuk menerima dan menyelesaikan *detail* pesanan. Isi status penyajian terdiri dari “belum diterima”, “lagi dimasak”, “sudah selesai”, dan “sudah disajikan”

sudah disajikan	Terima Selesai
lagi dimasak	Terima Selesai

Gambar 4.189 Tampilan status *detail* pesanan pada halaman *dashboard* dapur



Gambar 4.190 Tampilan notifikasi pesanan belum diterima pada halaman *dashboard* dapur



Gambar 4.191 Tampilan notifikasi pesanan sudah diterima pada halaman *dashboard* dapur

Ketika menekan tombol “Terima”, status pesanan akan berubah dari “belum diterima” menjadi “lagi dimasak”. Ketika menekan tombol “Selesai”, status pesanan akan berubah dari “lagi dimasak” menjadi “sudah selesai”.

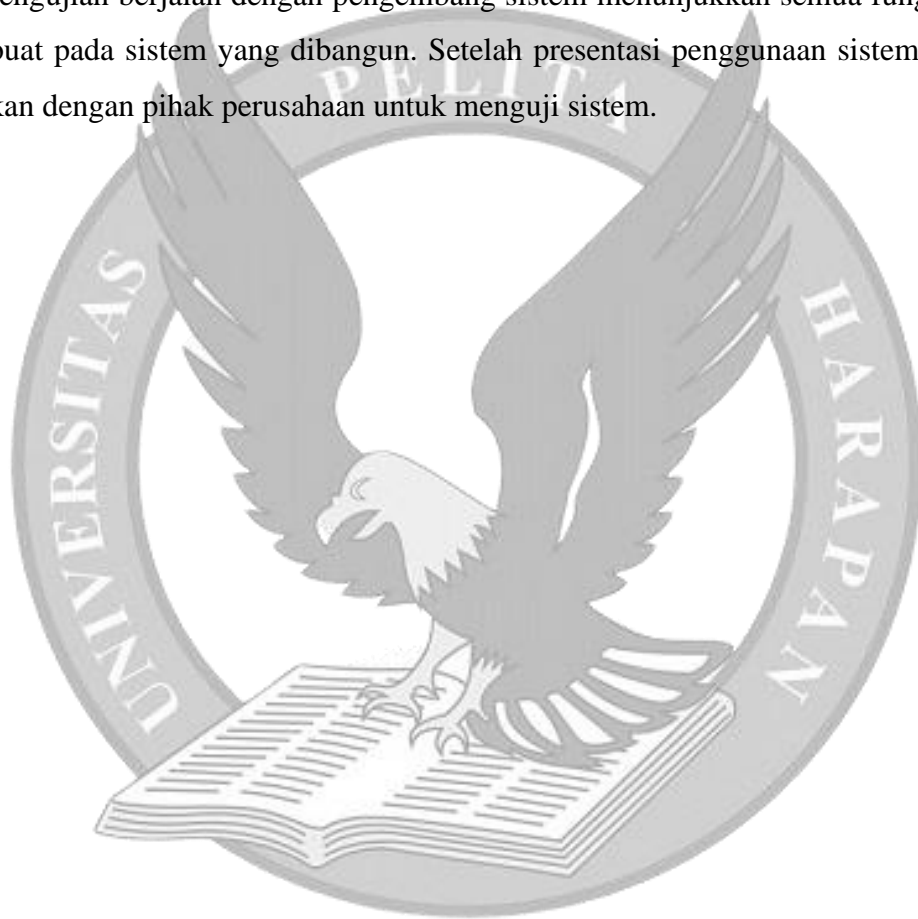
Jika status pesanan adalah “belum diterima” ketika menekan “Selesai”, maka sistem akan menampilkan notifikasi pesanan belum diterima.

Jika status pesanan adalah “lagi dimasak” ketika menekan “Terima”, maka sistem akan menampilkan notifikasi pesanan sudah diterima.

4.4.3. Tahap Pengujian

Tahap pengujian perlu dilakukan untuk memastikan setiap fungsi yang telah dibangun dalam sistem bekerja dengan apa yang diinginkan. Fungsi-fungsi yang akan diuji harus mencakup apa yang diinginkan oleh pihak perusahaan. Metode yang digunakan pada tahap pengujian ini adalah metode *black box*, dengan penekanan pada *scenario* dan pengujian *function*.

Pengujian dilakukan pada tanggal 12 Desember 2022 secara *on-site*. Pengujian dilakukan dan didampingi oleh pengembang sistem dan manajer Restoran Ta Chia. Proses pengujian berjalan dengan pengembang sistem menunjukkan semua fungsi yang telah dibuat pada sistem yang dibangun. Setelah presentasi penggunaan sistem selesai, dilanjutkan dengan pihak perusahaan untuk menguji sistem.



Berikut adalah salah satu *scenario* pada tahap pengujian, yaitu *scenario login*. Untuk pengujian *scenario* lainnya dapat dilihat pada lampiran bagian *user acceptance test*.

SCENARIO #1: LOGIN

Scenario Description

Skenario ini menjelaskan mengenai proses *user* melakukan *log in* ke dalam sistem

Test Scripts

Skenario ini terdiri dari *test script(s)* berikut:

- 1) *Log In user*

Use Case

- 1) *Log In*
- 2) *Log out*

User Group

- 1) Manajer
- 2) Kasir
- 3) Dapur

Script #1.1: Log In

Script Description

Test script ini menjelaskan proses *user* melakukan *log in* ke dalam sistem

Setup

- 1) *User* membuka halaman *log in*



Script Steps

Tabel 4.18 Script #1.1 Log In

Step #	Test Action	Expected Results	Status	Reference
1	User menekan tombol <i>log in</i> tanpa mengisi nama dan <i>password</i>	Sistem menampilkan notifikasi nama atau <i>password</i> salah	<i>Pass</i>	Gambar 4.80
2	User mengisi nama dan <i>password</i> yang tidak terdaftar dan menekan tombol <i>log in</i>	Sistem menampilkan notifikasi nama atau <i>password</i> salah	<i>Pass</i>	Gambar 4.80
3	User mengisi <i>password</i> yang salah dan menekan tombol <i>log in</i>	Sistem menampilkan notifikasi <i>password</i> salah	<i>Pass</i>	Gambar 4.81
4	User mengisi nama dan <i>password</i> pengguna dengan status <i>active</i>	Sistem menampilkan halaman <i>dashboard</i>	<i>Pass</i>	Gambar 4.86
5	User mengisi nama dan <i>password</i> pengguna dengan status <i>inactive</i>	Sistem menampilkan notifikasi <i>user</i> tidak dapat mengakses halaman	<i>Pass</i>	Gambar 4.82

Script #1.2: Log Out

Script Description

Test script ini menjelaskan proses *user* melakukan *log out* dari sistem

Setup

- 1) *User log in* membuka halaman *dashboard*

Script Steps

Tabel 4.19 Script #1.2 Log Out

Step #	Test Action	Expected Results	Status	Reference
1	User menekan tombol <i>log out</i>	Sistem menampilkan notifikasi konfirmasi <i>log out</i>	<i>Pass</i>	Gambar 4.88
2	User menekan tombol "OK" pada konfirmasi <i>log out</i>	Sistem menampilkan halaman <i>log in</i>	<i>Pass</i>	Gambar 4.79

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Dari penelitian yang dilakukan pada Restoran Ta Chia, diketahui bahwa proses pencatatan data-data dan informasi yang tersimpan masih dilakukan secara manual. Proses pemesanan pelanggan dimana seorang pelayan harus berjalan bolak-balik dari kasir dan dapur menghambat efisiensi pengambilan pesanan. Selain itu juga dengan pencatatan data-data restoran yang masih menggunakan kertas dan buku. Hal tersebut membuat pencatatan restoran dapat menjadi tidak akurat dan memberikan beban yang lebih berat terutama pada perhitungan stok barang dan keuangan.

Dengan memperhitungkan masalah dan kekurangan yang ada restoran, terancangnya sistem yang dapat membantu mengurangi beban pekerjaan mereka. Hasil dari pembuatan Tugas Akhir ini adalah sebuah *website* yang dapat mencatat data keuangan, seperti pemasukan dan pengeluaran, dan data stok inventori barang. Selain penyimpanan data berupa keuangan dan inventori, ada juga fitur untuk menyimpan data-data lain seperti karyawan, *role* pegawai, *menu* makanan, *supplier*, meja makan, dan pelanggan. Selain itu, *website* juga menyediakan fungsi yang menghubungkan pihak kasir dengan dapur.

Berdasarkan masukan yang diberikan dari pihak restoran, sistem yang dirancang telah membantu pekerjaan mereka dari beberapa pihak. Pencatatan data menjadi lebih mudah untuk dilakukan dan meminimalisir kesalahan pencatatan, terutama dengan perhitungan stok inventori dan keuangan. Hal ini sangat membantu karena dengan sistem ini karyawan tidak perlu menghitung data-data dengan tulis tangan sehingga mengurangi beban kerja mereka. Selain itu juga dengan adanya fungsi yang menghubungi kasir dengan dapur, pekerjaan yang tidak diperlukan oleh para pelayan yang memberikan pesanan dari kasir ke pihak dapur tidak diperlukan. Dengan demikian, proses pencatatan restoran yang dilakukan secara manual telah dikomputerisasikan sehingga mempermudah pekerjaan dan mengurangi beban kerja restoran.

5.2 Saran

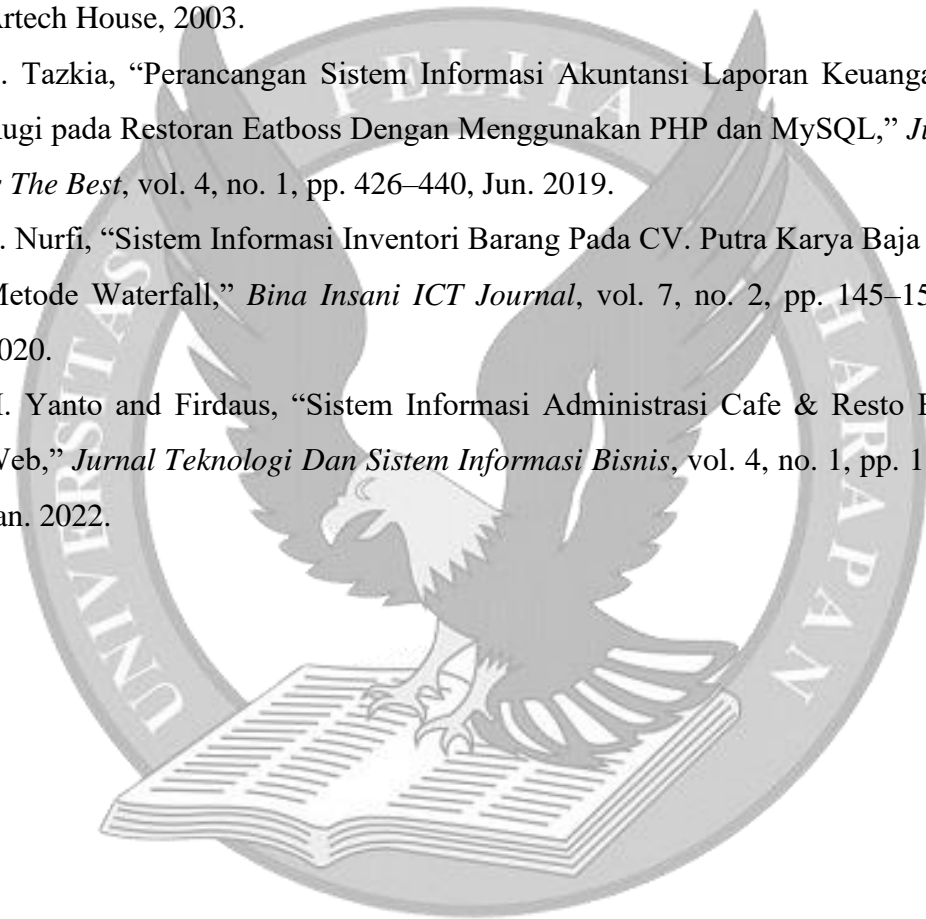
Sistem pencatatan keuangan dan inventori ini masih dapat dikembangkan lebih lanjut pada kemudian hari untuk menjadi lebih sempurna. Saran yang dapat diberikan kepada sistem ini adalah berikut:

- 1) Menambahkan fitur pada pengelolaan pesanan untuk meng-*input* jenis pesanan seperti *dine-in*, *take-away*, reservasi, dan penggunaan *gojek* atau *grab*.
- 2) Mempertimbangkan kemungkinan terjadinya pembatalan pesanan ketika seorang pelanggan tidak jadi membuat pesanan. Sistem dapat ditambahkan fungsi untuk membatalkan pesanan yang telah dibuat meskipun pemesanan *menu* belum dilakukan.
- 3) Menambahkan fitur yang dapat mengatur pengurangan stok bahan baku pada *menu* yang memiliki jumlah bahan yang bisa berbeda-beda pada tiap pesanan.
- 4) Mempertimbangkan penambahan fitur dimana sistem dapat melakukan perhitungan apa saja stok bahan baku yang paling diperlukan untuk dibeli berdasarkan jumlah *menu* yang dipesan.
- 5) Mengubah tampilan *bill* penagihan dan *bill* pelunasan menjadi lebih informatif atau mudah dimengerti.
- 6) Mengubah tampilan *UI/UX dashboard* kasir menjadi lebih baik. Pada saat pengembangan sistem, *dashboard* kasir menampilkan semua pesanan yang telah dibuat pada hari tertentu. *Dashboard* pesanan dapat diubah untuk menampilkan meja-meja restoran dengan status ketersediaannya daripada menampilkan setiap pesanan secara individu.
- 7) Mengembangkan sistem supaya dapat mengelola data-data pada bisnis *multi franchise*.

DAFTAR PUSTAKA

- [1] Mokolensang A, Pangemanan L, and Jocom S, “Analisis Keuntungan Penggunaan Teknologi Informasi Terhadap Usaha Kuliner Di Kecamatan Kauditan Kabupaten Minahasa Utara,” *Terakreditasi Jurnal Nasional Sinta 5*, vol. 17, no. 1, pp. 93–102, Jan. 2021.
- [2] R. Tarigan and H. Handiyani, “Manfaat Implementasi Dokumentasi Asuhan Keperawatan Berbasis Komputerisasi Dalam Meningkatkan Mutu Usaha Keperawatan,” *Jurnal Ilmiah Kesehatan Pencerah*, vol. 8, no. 2, pp. 110–116, Dec. 2019.
- [3] I.-B. Gustafsson, Å. Öström, J. Johansson, and L. Mossberg, “The Five Aspects Meal Model: a tool for developing meal services in restaurants,” *Blackwell Publishing Journal of Foodservice*, vol. 17, pp. 84–93, Jul. 2006.
- [4] G. Kurian and P. M. Muzumdar, “Restaurant Formality And Customer Service Dimensions In The Restaurant Industry: An Empirical Study,” *Atlantic Marketing Journal*, vol. 6, no. 1, pp. 75–92, Jun. 2017.
- [5] I. Nugroho, “Peranan Teknologi Informasi Dalam Audit Sistem Informasi Komputerisasi Akuntansi,” *Dinamika Informatika*, vol. 1, no. 2, pp. 122–130, Sep. 2009.
- [6] F. Rangkuti, *Manajemen Persediaan Aplikasi dibidang Bisnis*. Jakarta: Manajemen PT Raja Grafindo, 2004.
- [7] R. V. Palit, Y. Rindegan, and A. Lumenta, “Rancangan Sistem Informasi Keuangan Gereja Berbasis Web Di Jemaat GMIM Bukit Moria Malalayang,” *E-Journal Teknik Elektro dan Komputer*, vol. 4, no. 7, pp. 1–7, 2015.
- [8] S. Zulfiya and A. Subandoro, “Implementasi Laporan Arus Kas Sederhana Menggunakan Sistem Komputerisasi Di M3L Fried Chicken Prambon Sidoarjo,” *Akuntansiku*, vol. 1, no. 2, pp. 108–112, 2022.
- [9] S. Haryana, “Pengembangan Perangkat Lunak dengan Menggunakan PHP,” *Jurnal Computech & Bisnis*, vol. 2, no. 1, pp. 14–21, Jun. 2008.
- [10] D. Upton, *Improve your PHP coding productivity with the free compact open source MVC CodeIgniter framework!* Birmingham: Packt Publishing, 2007.
- [11] L. Ezell, *Practical CodeIgniter 3*. Lean Publishing, 2016.
- [12] S. Nurhidayah, M. Fauzan, and W. Rahayu, *Implementasi Metode Analytic Hierarchy Process (AHP) Dengan PHP*. Bandung: Kreatif Industri Nusantara, 2020.

- [13] J. Robbins, *Learning Web Design*, 4th ed. Canada: O'Reilly Media, Inc., 2012.
- [14] E. Setyawati, H. Sarwani, H. Wijoyo, and N. Soeharmoko, *Relational Database Management System (RDBMS)*. Jawa Tengah: Pena Persada, 2020.
- [15] C. Coronel, S. Morris, and P. Rob, *Database Principles: Fundamentals of Design, Implementation, and Management*, 10th ed. New York: Course Technology, Cengage Learning, 2013.
- [16] A. Dennis, B. Wixom, and D. Tegarden, *Systems Analysis and Design*, 5th ed. United States of America: Wiley, 2015.
- [17] J. Gao, *Testing and Quality Assurance for Component-based Software*. London: Artech House, 2003.
- [18] Z. Tazkia, "Perancangan Sistem Informasi Akuntansi Laporan Keuangan Laba Rugi pada Restoran Eatboss Dengan Menggunakan PHP dan MySQL," *Jurnal @ is The Best*, vol. 4, no. 1, pp. 426–440, Jun. 2019.
- [19] S. Nurfi, "Sistem Informasi Inventori Barang Pada CV. Putra Karya Baja Dengan Metode Waterfall," *Bina Insani ICT Journal*, vol. 7, no. 2, pp. 145–155, Dec. 2020.
- [20] H. Yanto and Firdaus, "Sistem Informasi Administrasi Cafe & Resto Berbasis Web," *Jurnal Teknologi Dan Sistem Informasi Bisnis*, vol. 4, no. 1, pp. 116–121, Jan. 2022.



LAMPIRAN A

HASIL WAWANCARA

A.1 Hasil Wawancara

Wawancara dilakukan sebanyak dua kali sebelum pembuatan sistem usulan dimulai untuk menentukan sistem yang diinginkan oleh pihak Restoran Ta Chia. Wawancara dilakukan secara langsung di Restoran Ta Chia. Narasumber yang hadir adalah pemilik restoran dan manajer restoran. Berikut merupakan hasil dari wawancara.

Q1: Siapa saja yang terlibat dalam proses pekerjaan pada sehari-hari?

A1: Yang paling terlibat adalah pihak kasir, pelayan, dan dapur. Kasir yang mengambil pesanan tamu dan mengatur mesin kasir buat pembayaran sekalian bon. Pelayan yang mengambil order tamu sama yang antar makanan/minuman ke tamu. Dapur yang masak makanan di bagian belakang.

Q2: Kalau untuk manajer? Apa saja yang dilakukan oleh manajer dalam proses pekerjaan?

A2: Manajer pada biasanya mengatur para pegawai yang bekerja di tempat. Dan kadang kala ada saat dimana manajer bisa membantu pekerjaan pelayan seperti mengambil pesanan tamu dan mengatur kasir. Manajer juga bertanggung jawab untuk mencatat stok sisa bahan baku setiap hari. Begitu juga dengan mengumpulkan bon-bon pemesanan dan mencatat pemasukan dan pengeluaran supaya dapat membantu pemilik melakukan perhitungan laba rugi.

Q3: Proses kerjanya sehari-harinya bagaimana ya? Dari ambil pesanan tamu sampai pembayaran.

A3: Pertama-tama kan tamu masuk, duduk di meja. Pelayan datang ambil pesanan tamu. Pelayan mencatat pesanan di note kecil. Catatannya dibagi menjadi 3 buat dibagi ke pihak kasir, dapur, dan tamu. Jadi pelayan harus jalan dari meja pelanggan, ke pihak kasir, lalu ke pihak dapur untuk mengasih notes-nya. Satu note ditaruh di meja tamu sebagai catatan pesanan apa saja yang belum diterima. Saat kasir dapat notes-nya, kasir masukin pesanan baru di mesin kasir. Saat dapur dapat notes-nya, dapur segera memasak pesannya. Setelah dimasak, pelayan bakal bawa masakannya dari pihak dapur ke meja tamu. Kalau tamu mau bayar, tamu ke kasir bayar lalu kasir mengeluarkan bon pembayaran yang dihasilkan dari mesin kasir.

Q4: Untuk pembayarannya dan pengeluaran uangnya dicatet dimana ya?

A4: Sekarang kami masih mencatat di buku besar. Manajer yang mencatat pemasukan pengeluaran, pemilik yang menghitung laba rugi. Untuk sekarang kami belum menggunakan komputer untuk mencatat semua data sepenuhnya. Tapi ada beberapa data yang kita sudah simpan dalam bentuk Excel, seperti bahan baku apa saja yang ada di tiap makanan, dan makanan dan minuman apa saja yang ada di menu kita. Meskipun begitu data yang kita simpan belum terisi dengan rapih, dan juga ada beberapa data yang kami masukkan terkadang kurang lengkap.

Q5: Untuk pencatatan sisa stok bahan baku apakah juga sama?

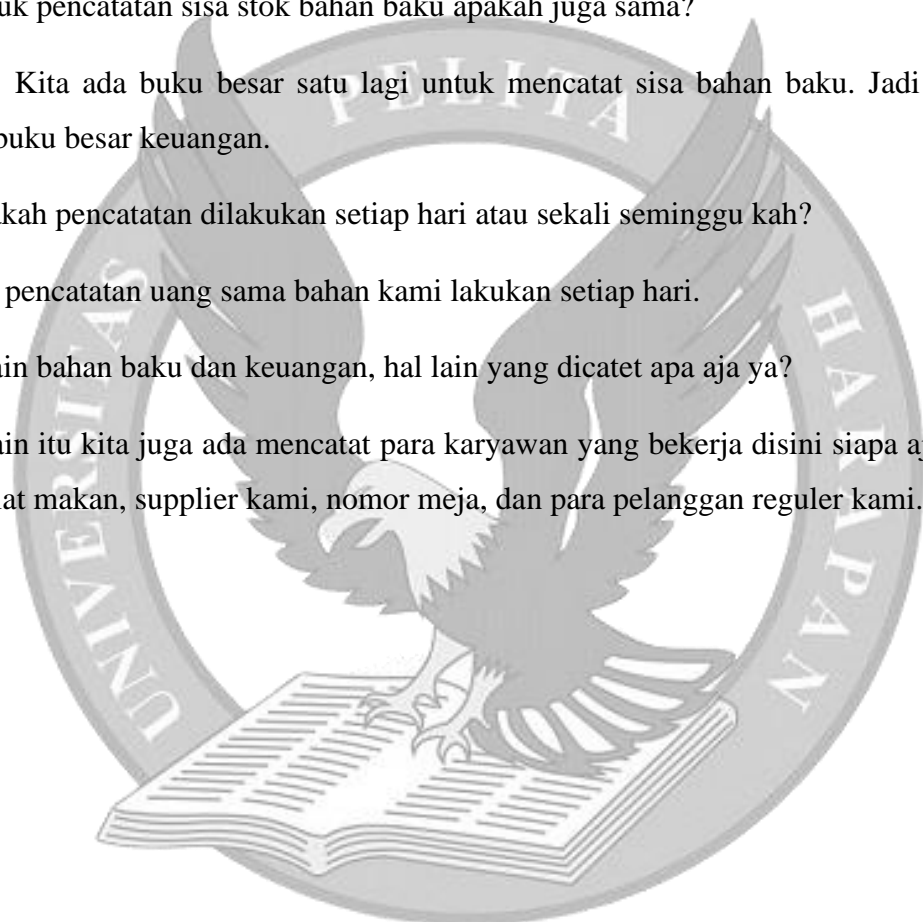
A5: Iya. Kita ada buku besar satu lagi untuk mencatat sisa bahan baku. Jadi dipisah dengan buku besar keuangan.

Q6: Apakah pencatatan dilakukan setiap hari atau sekali seminggu kah?

A6: Iya, pencatatan uang sama bahan kami lakukan setiap hari.

Q7: Selain bahan baku dan keuangan, hal lain yang dicatet apa aja ya?

A7: Selain itu kita juga ada mencatat para karyawan yang bekerja disini siapa aja, buku menu, alat makan, supplier kami, nomor meja, dan para pelanggan reguler kami.



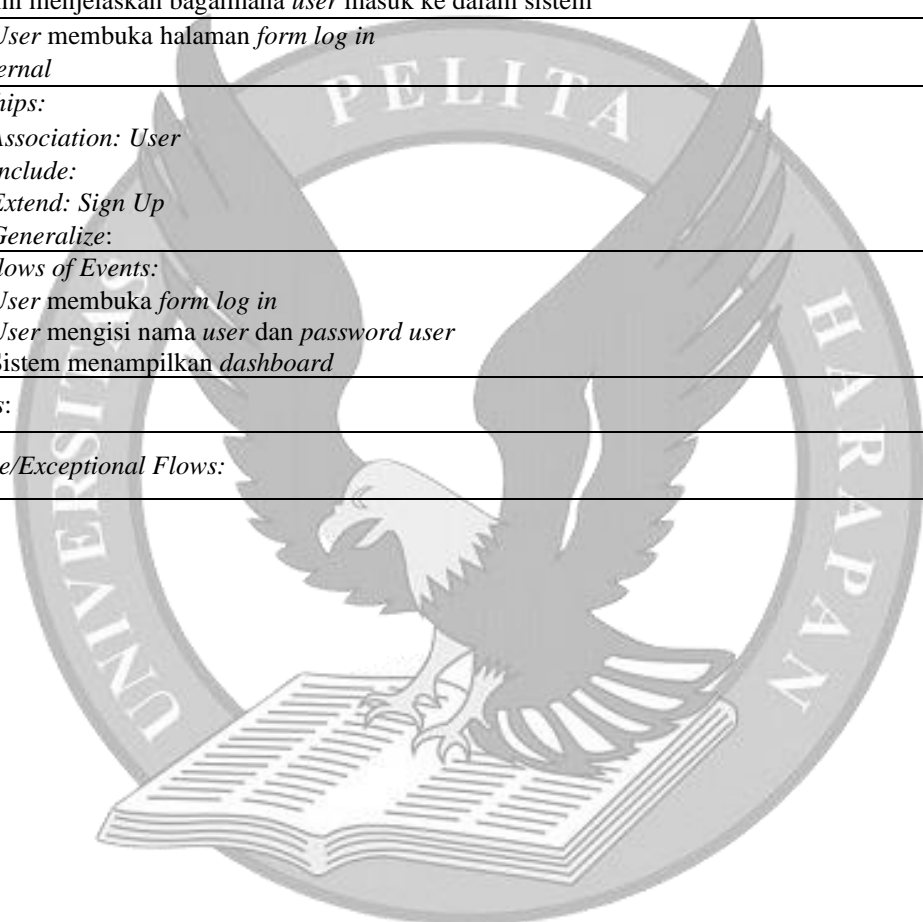
LAMPIRAN B

USE CASE DESCRIPTION

B.1 Log In

Tabel B.1 Use Case Description Log In

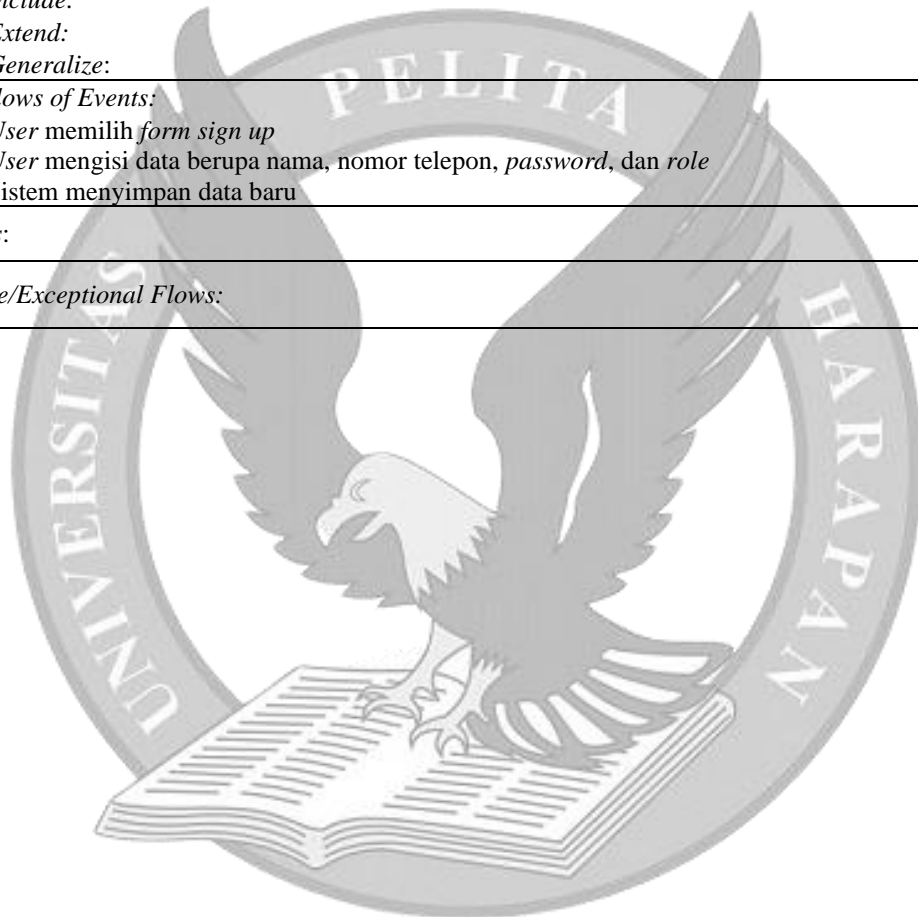
<i>Use Case Name: Log In</i>	ID: 1	<i>Importance Level: high</i>
<i>Primary Actor: User</i>	<i>Use Case Type: Detail, essential</i>	
<i>Stakeholders and Interests:</i> <i>User – Ingin mengakses sistem</i>		
<i>Brief Description:</i> <i>Use case ini menjelaskan bagaimana user masuk ke dalam sistem</i>		
<i>Trigger: User membuka halaman form log in</i> <i>Type: External</i>		
<i>Relationships:</i> <ul style="list-style-type: none"> • <i>Association: User</i> • <i>Include:</i> • <i>Extend: Sign Up</i> • <i>Generalize:</i> 		
<i>Normal Flows of Events:</i> <ol style="list-style-type: none"> 1. <i>User membuka form log in</i> 2. <i>User mengisi nama user dan password user</i> 3. <i>Sistem menampilkan dashboard</i> 		
<i>Sub Flows:</i>		
<i>Alternative/Exceptional Flows:</i>		



B.2 Sign Up

Tabel B.2 Use Case Description Sign Up

<i>Use Case Name: Sign Up</i>	<i>ID: 2</i>	<i>Importance Level: high</i>
<i>Primary Actor: User</i>	<i>Use Case Type: Detail, essential</i>	
<i>Stakeholders and Interests:</i> <i>User – Ingin mendapatkan akses ke dalam sistem</i>		
<i>Brief Description:</i> <i>Use case ini menjelaskan bagaimana user melakukan pendaftaran pengguna baru ke dalam sistem</i>		
<i>Trigger: User memilih halaman form sign up</i> <i>Type: External</i>		
<i>Relationships:</i> <ul style="list-style-type: none">• <i>Association: User</i>• <i>Include:</i>• <i>Extend:</i>• <i>Generalize:</i>		
<i>Normal Flows of Events:</i> <ol style="list-style-type: none">1. <i>User memilih form sign up</i>2. <i>User mengisi data berupa nama, nomor telepon, password, dan role</i>3. <i>Sistem menyimpan data baru</i>		
<i>Sub Flows:</i>		
<i>Alternative/Exceptional Flows:</i>		



B.3 Mengelola User

Tabel B.3 Use Case Description Mengelola User

<i>Use Case Name:</i> Mengelola User	ID: 3	<i>Importance Level:</i> high
<i>Primary Actor:</i> Manajer	<i>Use Case Type:</i> Detail, essential	
<i>Stakeholders and Interests:</i> Manajer – Ingin menambah atau mengubah data user		
<i>Brief Description:</i> Use case ini menjelaskan bagaimana manajer mengelola data user		
<i>Trigger:</i> Manajer memilih master data user <i>Type:</i> External		
<i>Relationships:</i> <ul style="list-style-type: none"> • <i>Association:</i> Manajer • <i>Include:</i> • <i>Extend:</i> • <i>Generalize:</i> 		
<i>Normal Flows of Events:</i> <ol style="list-style-type: none"> 1) Manajer memilih master data user 2) Sistem menampilkan master data user <ul style="list-style-type: none"> Jika manajer ingin menambah data baru S-1: Tambah data user Jika manajer ingin mengubah data S-2: Edit data user 3) Sistem menyimpan data 		
<i>Sub Flows:</i> <p>S-1: Tambah data user</p> <ol style="list-style-type: none"> 1) Manajer memasukkan data user baru <p>S-2: Edit data user</p> <ol style="list-style-type: none"> 1) Manajer mengubah data user 		
<i>Alternative/Exceptional Flows:</i>		

B.4 Mengelola *Role*

Tabel B.4 *Use Case Description* Mengelola *Role*

<i>Use Case Name:</i> Mengelola <i>Role</i>	ID: 4	<i>Importance Level:</i> high
<i>Primary Actor:</i> Manajer	<i>Use Case Type:</i> Detail, essential	
<i>Stakeholders and Interests:</i> Manajer – Ingin menambah atau mengubah data <i>role</i>		
<i>Brief Description:</i> <i>Use case ini menjelaskan bagaimana manajer mengelola data role</i>		
<i>Trigger:</i> Manajer memilih <i>master data role</i> <i>Type:</i> External		
<i>Relationships:</i> <ul style="list-style-type: none"> • <i>Association:</i> Manajer • <i>Include:</i> • <i>Extend:</i> • <i>Generalize:</i> 		
<i>Normal Flows of Events:</i> <ol style="list-style-type: none"> 1) Manajer memilih <i>master data role</i> 2) Sistem menampilkan <i>master data role</i> <ul style="list-style-type: none"> Jika manajer ingin menambah data baru S-1: Tambah data <i>role</i> Jika manajer ingin mengubah data S-2: <i>Edit data role</i> 3) Sistem menyimpan data 		
<i>Sub Flows:</i> <p>S-1: Tambah data <i>role</i></p> <ol style="list-style-type: none"> 1) Manajer memasukkan data <i>role</i> baru <p>S-2: <i>Edit data role</i></p> <ol style="list-style-type: none"> 1) Manajer mengubah data <i>role</i> 		
<i>Alternative/Exceptional Flows:</i>		

B.5 Mengelola Menu

Tabel B.5 Use Case Description Mengelola Menu

<i>Use Case Name:</i> Mengelola Menu	ID: 5	<i>Importance Level:</i> high
<i>Primary Actor:</i> Manajer	<i>Use Case Type:</i> Detail, essential	
<i>Stakeholders and Interests:</i> Manajer – Ingin menambah atau mengubah data menu		
<i>Brief Description:</i> Use case ini menjelaskan bagaimana manajer mengelola data menu		
<i>Trigger:</i> Manajer memilih master data menu <i>Type:</i> External		
<i>Relationships:</i> <ul style="list-style-type: none"> • <i>Association:</i> Manajer • <i>Include:</i> • <i>Extend:</i> • <i>Generalize:</i> 		
<i>Normal Flows of Events:</i> <ol style="list-style-type: none"> 1) Manajer memilih master data menu 2) Sistem menampilkan master data menu <ul style="list-style-type: none"> Jika manajer ingin menambah data baru S-1: Tambah data menu Jika manajer ingin mengubah data S-2: Edit data menu 3) Sistem menyimpan data 		
<i>Sub Flows:</i> <p>S-1: Tambah data menu</p> <ol style="list-style-type: none"> 1) Manajer memasukkan data menu baru <p>S-2: Edit data menu</p> <ol style="list-style-type: none"> 1) Manajer mengubah data menu 		
<i>Alternative/Exceptional Flows:</i>		

B.6 Mengelola *Detail Menu*

Tabel B.6 *Use Case Description* Mengelola *Detail Menu*

<i>Use Case Name:</i> Mengelola <i>Detail Menu</i>	ID: 6	<i>Importance Level:</i> high
<i>Primary Actor:</i> Manajer	<i>Use Case Type:</i> Detail, essential	
<i>Stakeholders and Interests:</i> Manajer – Ingin menambah atau mengubah data <i>menu detail</i>		
<i>Brief Description:</i> <i>Use case ini menjelaskan bagaimana manajer mengelola data menu detail</i>		
<i>Trigger:</i> Manajer memilih <i>master data menu detail</i> <i>Type:</i> External		
<i>Relationships:</i> <ul style="list-style-type: none"> • <i>Association:</i> Manajer • <i>Include:</i> • <i>Extend:</i> • <i>Generalize:</i> 		
<i>Normal Flows of Events:</i> <ol style="list-style-type: none"> 1) Manajer memilih <i>master data menu detail</i> dari <i>master data menu</i> 2) Sistem menampilkan <i>master data menu detail</i> <ul style="list-style-type: none"> Jika manajer ingin menambah data baru S-1: Tambah data <i>menu detail</i> Jika manajer ingin mengubah data S-2: <i>Edit data menu detail</i> 3) Sistem menyimpan data 		
<i>Sub Flows:</i> <p>S-1: Tambah data <i>menu detail</i></p> <ol style="list-style-type: none"> 1) Manajer memasukkan data <i>menu detail</i> baru <p>S-2: <i>Edit data menu detail</i></p> <ol style="list-style-type: none"> 1) Manajer mengubah data <i>menu detail</i> 		
<i>Alternative/Exceptional Flows:</i>		

B.7 Mengelola Kategori Menu

Tabel B.7 Use Case Description Mengelola Kategori Menu

<i>Use Case Name:</i> Mengelola Kategori Menu	ID: 7	<i>Importance Level:</i> high
<i>Primary Actor:</i> Manajer	<i>Use Case Type:</i> Detail, essential	
<i>Stakeholders and Interests:</i> Manajer – Ingin menambah atau mengubah data kategori menu		
<i>Brief Description:</i> Use case ini menjelaskan bagaimana manajer mengelola data kategori menu		
<i>Trigger:</i> Manajer memilih master data kategori menu <i>Type:</i> External		
<i>Relationships:</i> <ul style="list-style-type: none"> • <i>Association:</i> Manajer • <i>Include:</i> • <i>Extend:</i> • <i>Generalize:</i> 		
<i>Normal Flows of Events:</i> <ol style="list-style-type: none"> 1) Manajer memilih master data kategori menu 2) Sistem menampilkan master data kategori menu <ul style="list-style-type: none"> Jika manajer ingin menambah data baru S-1: Tambah data kategori menu Jika manajer ingin mengubah data S-2: Edit data kategori menu 3) Sistem menyimpan data 		
<i>Sub Flows:</i> <p>S-1: Tambah data kategori menu</p> <ol style="list-style-type: none"> 1) Manajer memasukkan data kategori menu baru <p>S-2: Edit data kategori menu</p> <ol style="list-style-type: none"> 1) Manajer mengubah data kategori menu 		
<i>Alternative/Exceptional Flows:</i>		

B.8 Mengelola Inventori Barang

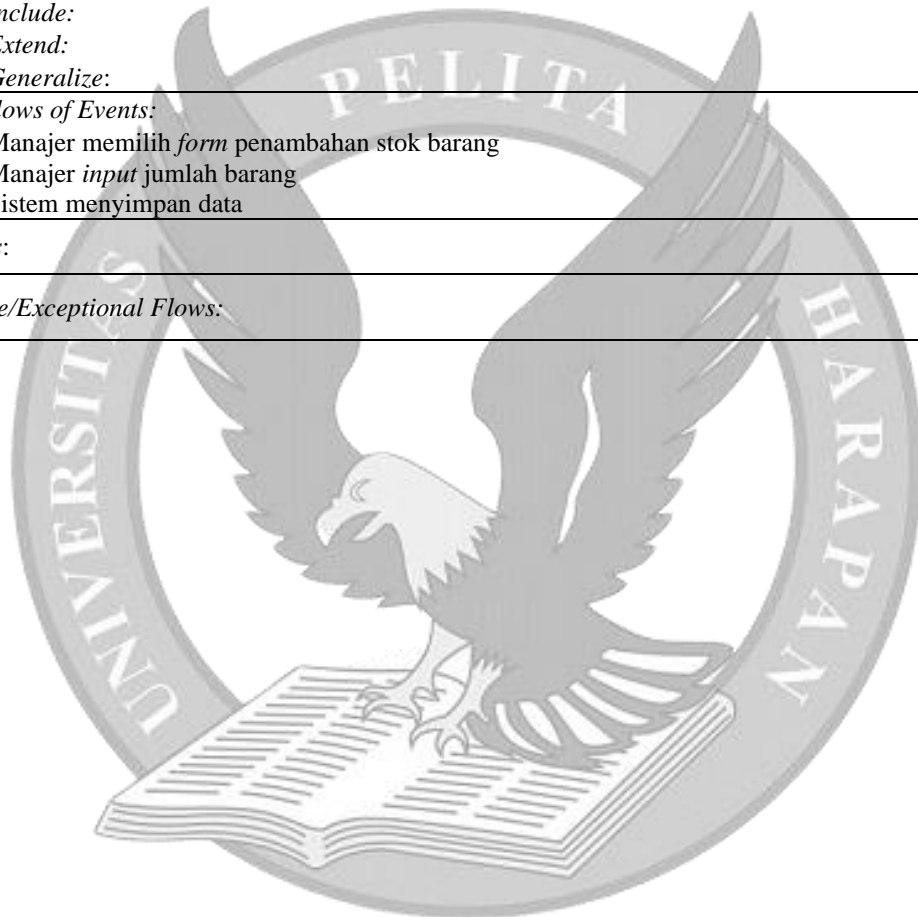
Tabel B.8 *Use Case Description* Inventori Barang

<i>Use Case Name:</i> Mengelola Inventori Barang	ID: 8	<i>Importance Level:</i> high
<i>Primary Actor:</i> Manajer	<i>Use Case Type:</i> Detail, essential	
<i>Stakeholders and Interests:</i> Manajer – Ingin menambah atau mengubah data inventori		
<i>Brief Description:</i> Use case ini menjelaskan bagaimana manajer mengelola data inventori		
<i>Trigger:</i> Manajer memilih <i>master data</i> inventori <i>Type:</i> External		
<i>Relationships:</i> <ul style="list-style-type: none"> • <i>Association:</i> Manajer • <i>Include:</i> • <i>Extend:</i> • <i>Generalize:</i> 		
<i>Normal Flows of Events:</i> <ol style="list-style-type: none"> 1) Manajer memilih <i>master data</i> inventori 2) Sistem menampilkan <i>master data</i> inventori <ul style="list-style-type: none"> Jika manajer ingin menambah data baru S-1: Tambah data inventori Jika manajer ingin mengubah data S-2: <i>Edit</i> data inventori 3) Sistem menyimpan data 		
<i>Sub Flows:</i> <p>S-1: Tambah data inventori</p> <ol style="list-style-type: none"> 1) Manajer memasukkan data inventori baru <p>S-2: <i>Edit</i> data inventori</p> <ol style="list-style-type: none"> 1) Manajer mengubah data inventori 		
<i>Alternative/Exceptional Flows:</i>		

B.9 Mengelola Stok Inventori Barang

Tabel B.9 *Use Case Description* Mengelola Stok Inventori Barang

<i>Use Case Name:</i> Mengelola Stok Inventori Barang	ID: 9	<i>Importance Level:</i> medium
<i>Primary Actor:</i> Manajer	<i>Use Case Type:</i> Detail, essential	
<i>Stakeholders and Interests:</i> Manajer – Ingin menambah stok inventori barang		
<i>Brief Description:</i> Use case ini menjelaskan bagaimana manajer menambah stok inventori barang		
<i>Trigger:</i> Stok inventori barang kurang <i>Type:</i> External		
<i>Relationships:</i> <ul style="list-style-type: none"> • <i>Association:</i> Manajer • <i>Include:</i> • <i>Extend:</i> • <i>Generalize:</i> 		
<i>Normal Flows of Events:</i> <ol style="list-style-type: none"> 1) Manajer memilih <i>form</i> penambahan stok barang 2) Manajer <i>input</i> jumlah barang 3) Sistem menyimpan data 		
<i>Sub Flows:</i>		
<i>Alternative/Exceptional Flows:</i>		



B.10 Mengelola Jenis Barang

Tabel B.10 *Use Case Description* Mengelola Jenis Barang

<i>Use Case Name:</i> Mengelola Jenis Barang	ID: 10	<i>Importance Level:</i> high
<i>Primary Actor:</i> Manajer	<i>Use Case Type:</i> Detail, essential	
<i>Stakeholders and Interests:</i> Manajer – Ingin menambah atau mengubah data jenis barang		
<i>Brief Description:</i> Use case ini menjelaskan bagaimana manajer mengelola data jenis barang		
<i>Trigger:</i> Manajer memilih <i>master data</i> jenis barang <i>Type:</i> External		
<i>Relationships:</i> <ul style="list-style-type: none"> • <i>Association:</i> Manajer • <i>Include:</i> • <i>Extend:</i> • <i>Generalize:</i> 		
<i>Normal Flows of Events:</i> <ol style="list-style-type: none"> 1) Manajer memilih <i>master data</i> jenis barang 2) Sistem menampilkan <i>master data</i> jenis barang <ul style="list-style-type: none"> Jika manajer ingin menambah data baru S-1: Tambah data jenis barang Jika manajer ingin mengubah data S-2: <i>Edit</i> data jenis barang 3) Sistem menyimpan data 		
<i>Sub Flows:</i> <p>S-1: Tambah data jenis barang</p> <ol style="list-style-type: none"> 1) Manajer memasukkan data jenis barang baru <p>S-2: <i>Edit</i> data jenis barang</p> <ol style="list-style-type: none"> 1) Manajer mengubah data jenis barang 		
<i>Alternative/Exceptional Flows:</i>		

B.11 Mengelola Satuan Barang

Tabel B.11 *Use Case Description* Mengelola Satuan Barang

<i>Use Case Name:</i> Mengelola Satuan Barang	ID: 11	<i>Importance Level:</i> high
<i>Primary Actor:</i> Manajer	<i>Use Case Type:</i> Detail, essential	
<i>Stakeholders and Interests:</i> Manajer – Ingin menambah atau mengubah data satuan		
<i>Brief Description:</i> Use case ini menjelaskan bagaimana manajer mengelola data satuan		
<i>Trigger:</i> Manajer memilih <i>master data</i> satuan <i>Type:</i> External		
<i>Relationships:</i> <ul style="list-style-type: none"> • <i>Association:</i> Manajer • <i>Include:</i> • <i>Extend:</i> • <i>Generalize:</i> 		
<i>Normal Flows of Events:</i> <ol style="list-style-type: none"> 1) Manajer memilih <i>master data</i> satuan 2) Sistem menampilkan <i>master data</i> satuan <ul style="list-style-type: none"> Jika manajer ingin menambah data baru S-1: Tambah data satuan Jika manajer ingin mengubah data S-2: <i>Edit</i> data satuan 3) Sistem menyimpan data 		
<i>Sub Flows:</i> <p>S-1: Tambah data satuan</p> <ol style="list-style-type: none"> 1) Manajer memasukkan data satuan baru <p>S-2: <i>Edit</i> data satuan</p> <ol style="list-style-type: none"> 1) Manajer mengubah data satuan 		
<i>Alternative/Exceptional Flows:</i>		

B.12 Mengelola Pemasukan

Tabel B.12 *Use Case Description* Mengelola Pemasukan

<i>Use Case Name:</i> Mengelola Pemasukan	ID: 12	<i>Importance Level:</i> high
<i>Primary Actor:</i> Manajer	<i>Use Case Type:</i> Detail, essential	
<i>Stakeholders and Interests:</i> Manajer – Ingin melihat data pemasukan		
<i>Brief Description:</i> Use case ini menjelaskan bagaimana manajer mengelola data pemasukan		
<i>Trigger:</i> Manajer memilih <i>master data</i> pemasukan <i>Type:</i> External		
<i>Relationships:</i> <ul style="list-style-type: none"> • <i>Association:</i> Manajer • <i>Include:</i> • <i>Extend:</i> • <i>Generalize:</i> 		
<i>Normal Flows of Events:</i> <ol style="list-style-type: none"> 1) Manajer memilih <i>master data</i> pemasukan 2) Sistem menampilkan <i>master data</i> pemasukan 3) Manajer pilih tanggal untuk <i>sorting</i> 4) Sistem menampilkan data pemasukan hasil <i>sorting</i> 		
<i>Sub Flows:</i>		
<i>Alternative/Exceptional Flows:</i>		



B.13 Mengelola Meja

Tabel B.13 *Use Case Description* Mengelola Meja

<i>Use Case Name:</i> Mengelola Meja	ID: 13	<i>Importance Level:</i> high
<i>Primary Actor:</i> Manajer	<i>Use Case Type:</i> Detail, essential	
<i>Stakeholders and Interests:</i> Manajer – Ingin menambah atau mengubah data meja		
<i>Brief Description:</i> Use case ini menjelaskan bagaimana manajer mengelola data meja		
<i>Trigger:</i> Manajer memilih <i>master data</i> meja <i>Type:</i> External		
<i>Relationships:</i> <ul style="list-style-type: none"> • <i>Association:</i> Manajer • <i>Include:</i> • <i>Extend:</i> • <i>Generalize:</i> 		
<i>Normal Flows of Events:</i> <ol style="list-style-type: none"> 1) Manajer memilih <i>master data</i> meja 2) Sistem menampilkan <i>master data</i> meja <ul style="list-style-type: none"> Jika manajer ingin menambah data baru S-1: Tambah data meja Jika manajer ingin mengubah data S-2: <i>Edit</i> data meja 3) Sistem menyimpan data 		
<i>Sub Flows:</i> <p>S-1: Tambah data meja</p> <ol style="list-style-type: none"> 1) Manajer memasukkan data meja baru <p>S-2: <i>Edit</i> data meja</p> <ol style="list-style-type: none"> 1) Manajer mengubah data meja 		
<i>Alternative/Exceptional Flows:</i>		

B.14 Mengelola Metode Pembayaran

Tabel B.14 *Use Case Description* Mengelola Metode Pembayaran

<i>Use Case Name:</i> Mengelola Metode Pembayaran	ID: 14	<i>Importance Level:</i> high
<i>Primary Actor:</i> Manajer	<i>Use Case Type:</i> Detail, essential	
<i>Stakeholders and Interests:</i> Manajer – Ingin menambah atau mengubah data metode bayar		
<i>Brief Description:</i> Use case ini menjelaskan bagaimana manajer mengelola data metode bayar		
<i>Trigger:</i> Manajer memilih <i>master data</i> metode bayar <i>Type:</i> External		
<i>Relationships:</i> <ul style="list-style-type: none"> • <i>Association:</i> Manajer • <i>Include:</i> • <i>Extend:</i> • <i>Generalize:</i> 		
<i>Normal Flows of Events:</i> <ol style="list-style-type: none"> 1) Manajer memilih <i>master data</i> metode bayar 2) Sistem menampilkan <i>master data</i> metode bayar <ul style="list-style-type: none"> Jika manajer ingin menambah data baru S-1: Tambah data metode bayar Jika manajer ingin mengubah data S-2: <i>Edit</i> data metode bayar 3) Sistem menyimpan data 		
<i>Sub Flows:</i> <p>S-1: Tambah data metode bayar</p> <ol style="list-style-type: none"> 1) Manajer memasukkan data metode bayar baru <p>S-2: <i>Edit</i> data metode bayar</p> <ol style="list-style-type: none"> 1) Manajer mengubah data metode bayar 		
<i>Alternative/Exceptional Flows:</i>		

B.15 Mengelola Pelanggan

Tabel B.15 *Use Case Description* Mengelola Pelanggan

<i>Use Case Name:</i> Mengelola Pelanggan	ID: 15	<i>Importance Level:</i> high
<i>Primary Actor:</i> Manajer	<i>Use Case Type:</i> Detail, essential	
<i>Stakeholders and Interests:</i> Manajer – Ingin menambah atau mengubah data metode bayar		
<i>Brief Description:</i> Use case ini menjelaskan bagaimana manajer mengelola data metode bayar		
<i>Trigger:</i> Manajer memilih <i>master data</i> metode bayar <i>Type:</i> External		
<i>Relationships:</i> <ul style="list-style-type: none"> • <i>Association:</i> Manajer • <i>Include:</i> • <i>Extend:</i> • <i>Generalize:</i> 		
<i>Normal Flows of Events:</i> <ol style="list-style-type: none"> 1) Manajer memilih <i>master data</i> metode bayar 2) Sistem menampilkan <i>master data</i> metode bayar <ul style="list-style-type: none"> Jika manajer ingin menambah data baru S-1: Tambah data metode bayar Jika manajer ingin mengubah data S-2: <i>Edit</i> data metode bayar 3) Sistem menyimpan data 		
<i>Sub Flows:</i> <p>S-1: Tambah data metode bayar</p> <ol style="list-style-type: none"> 1) Manajer memasukkan data metode bayar baru <p>S-2: <i>Edit</i> data metode bayar</p> <ol style="list-style-type: none"> 1) Manajer mengubah data metode bayar 		
<i>Alternative/Exceptional Flows:</i>		

B.16 Mengelola Pengeluaran

Tabel B.16 *Use Case Description* Mengelola Pengeluaran

<i>Use Case Name:</i> Mengelola Pengeluaran	ID: 16	<i>Importance Level:</i> high
<i>Primary Actor:</i> Manajer	<i>Use Case Type:</i> Detail, essential	
<i>Stakeholders and Interests:</i> Manajer – Ingin menambah atau mengubah data pengeluaran		
<i>Brief Description:</i> Use case ini menjelaskan bagaimana manajer mengelola data pengeluaran		
<i>Trigger:</i> Manajer memilih <i>master data</i> pengeluaran <i>Type:</i> External		
<i>Relationships:</i> <ul style="list-style-type: none"> • <i>Association:</i> Manajer • <i>Include:</i> • <i>Extend:</i> • <i>Generalize:</i> 		
<i>Normal Flows of Events:</i> <ol style="list-style-type: none"> 1) Manajer memilih <i>master data</i> pengeluaran 2) Sistem menampilkan <i>master data</i> pengeluaran <ul style="list-style-type: none"> Jika manajer ingin menambah data baru S-1: Tambah data pengeluaran Jika manajer ingin mengubah data S-2: <i>Edit</i> data pengeluaran 3) Sistem menyimpan data 		
<i>Sub Flows:</i> <p>S-1: Tambah data pengeluaran</p> <ol style="list-style-type: none"> 1) Manajer memasukkan data pengeluaran baru <p>S-2: <i>Edit</i> data pengeluaran</p> <ol style="list-style-type: none"> 1) Manajer mengubah data pengeluaran 		
<i>Alternative/Exceptional Flows:</i>		

B.17 Mengelola *Supplier*

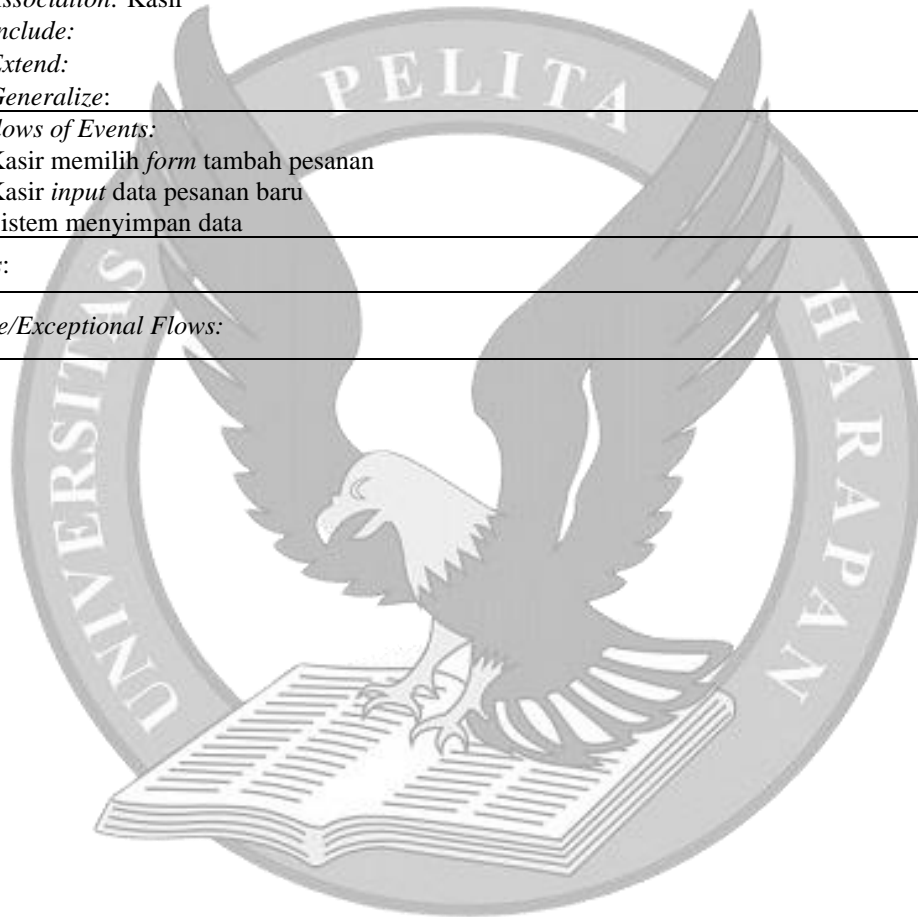
Tabel B.17 *Use Case Description* Mengelola *Supplier*

<i>Use Case Name:</i> Mengelola <i>Supplier</i>	ID: 17	<i>Importance Level:</i> high
<i>Primary Actor:</i> Manajer	<i>Use Case Type:</i> Detail, essential	
<i>Stakeholders and Interests:</i> Manajer – Ingin menambah atau mengubah data <i>supplier</i>		
<i>Brief Description:</i> <i>Use case ini menjelaskan bagaimana manajer mengelola data supplier</i>		
<i>Trigger:</i> Manajer memilih <i>master data supplier</i> <i>Type:</i> External		
<i>Relationships:</i> <ul style="list-style-type: none">• <i>Association:</i> Manajer• <i>Include:</i>• <i>Extend:</i>• <i>Generalize:</i>		
<i>Normal Flows of Events:</i> <ol style="list-style-type: none">1) Manajer memilih <i>master data supplier</i>2) Sistem menampilkan <i>master data supplier</i><ul style="list-style-type: none">Jika manajer ingin menambah data baru S-1: Tambah data <i>supplier</i>Jika manajer ingin mengubah data S-2: <i>Edit data supplier</i>3) Sistem menyimpan data		
<i>Sub Flows:</i> S-1: Tambah data <i>supplier</i> <ol style="list-style-type: none">1) Manajer memasukkan data <i>supplier</i> baru S-2: <i>Edit data supplier</i> <ol style="list-style-type: none">1) Manajer mengubah data <i>supplier</i>		
<i>Alternative/Exceptional Flows:</i>		

B.18 Mengelola Pesanan Pelanggan

Tabel B.18 *Use Case Description* Mengelola Pesanan Pelanggan

<i>Use Case Name:</i> Mengelola Pesanan Pelanggan	ID: 18	<i>Importance level:</i> high
<i>Primary Actor:</i> Kasir	<i>Use Case Type:</i> Detail, essential	
<i>Stakeholders and Interests:</i> Kasir - Ingin menambah data pesanan pelanggan Pelanggan – Ingin membuat pesanan		
<i>Brief Description:</i> Use case ini menjelaskan bagaimana kasir menambah data pesanan pelanggan		
<i>Trigger:</i> Pelanggan membuat pesanan <i>Type:</i> External		
<i>Relationships:</i> <ul style="list-style-type: none"> • <i>Association:</i> Kasir • <i>Include:</i> • <i>Extend:</i> • <i>Generalize:</i> 		
<i>Normal Flows of Events:</i> <ol style="list-style-type: none"> 1) Kasir memilih <i>form</i> tambah pesanan 2) Kasir <i>input</i> data pesanan baru 3) Sistem menyimpan data 		
<i>Sub Flows:</i>		
<i>Alternative/Exceptional Flows:</i>		



B.19 Mengelola *Detail* Pesanan Pelanggan

Tabel B.19 *Use Case Description* Mengelola *Detail* Pesanan Pelanggan

<i>Use Case Name:</i> Mengelola <i>Detail</i> Pesanan Pelanggan	ID: 19	<i>Importance level:</i> high
<i>Primary Actor:</i> Kasir	<i>Use Case Type:</i> Detail, essential	
<i>Stakeholders and Interests:</i> Kasir – Ingin menambah data pesanan <i>detail</i>		
<i>Brief Description:</i> <i>Use case ini menjelaskan bagaimana kasir mengelola pesanan detail</i>		
<i>Trigger:</i> Menambahkan <i>menu</i> ke dalam pesanan <i>Type:</i> External		
<i>Relationships:</i> <ul style="list-style-type: none"> • <i>Association:</i> Kasir • <i>Include:</i> Melakukan Konfirmasi Pesanan • <i>Extend:</i> • <i>Generalize:</i> 		
<i>Normal Flows of Events:</i> <ol style="list-style-type: none"> 1) Kasir memilih data pesanan <i>detail</i> 2) Kasir memilih <i>form</i> penambahan <i>menu</i> 3) Kasir <i>input menu</i> ke dalam pesanan <ul style="list-style-type: none"> Jika stok barang mencukupi S-1: Konfirmasi pesanan Jika stok barang tidak mencukupi S-2: Menampilkan notifikasi 		
<i>Sub Flows:</i> <p>S-1: Konfirmasi pesanan</p> <ol style="list-style-type: none"> 1) <i>Trigger case ID</i> 19 <p>S-2: Menampilkan notifikasi</p> <ol style="list-style-type: none"> 1) Sistem menampilkan notifikasi stok barang kurang 		
<i>Alternative/Exceptional Flows:</i>		

B.20 Melakukan Konfirmasi Pesanan

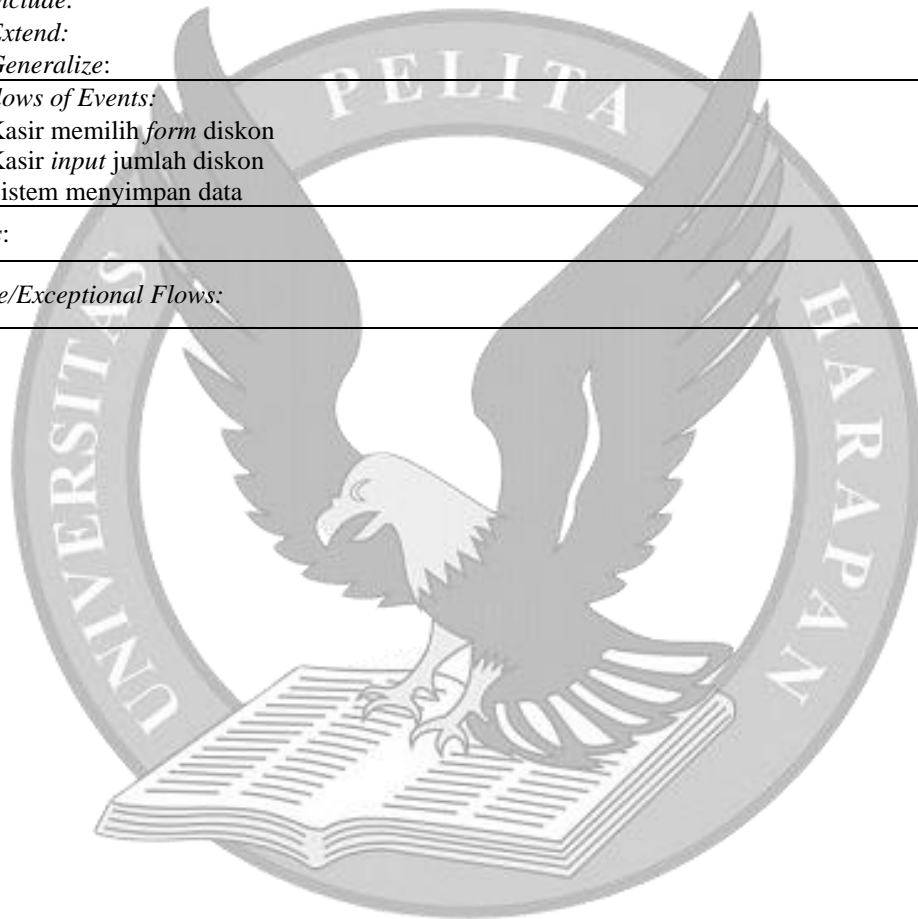
Tabel B.20 *Use Case Description* Melakukan Konfirmasi Pesanan

<i>Use Case Name:</i> Melakukan Konfirmasi Pesanan	ID: 20	<i>Importance level:</i> high
<i>Primary Actor:</i> Dapur	<i>Use Case Type:</i> Detail, essential	
<i>Stakeholders and Interests:</i> Kasir – Menambah <i>menu</i> ke dalam pesanan Dapur – Menerima pesanan baru		
<i>Brief Description:</i> <i>Use case</i> ini menjelaskan bagaimana dapur melakukan konfirmasi pesanan baru		
<i>Trigger:</i> Penambahan <i>menu</i> baru ke dalam pesanan <i>Type:</i> External		
<i>Relationships:</i> <ul style="list-style-type: none"> • <i>Association:</i> Dapur • <i>Include:</i> • <i>Extend:</i> • <i>Generalize:</i> 		
<i>Normal Flows of Events:</i> <ol style="list-style-type: none"> 1) Dapur memilih <i>dashboard</i> dapur 2) Dapur menerima pesanan baru <ul style="list-style-type: none"> Jika pesanan di-cancel S-1: Menampilkan notifikasi Jika pesanan belum di-cancel S-2: Buat pesanan 		
<i>Sub Flows:</i> <p>S-1: Menampilkan notifikasi</p> <ol style="list-style-type: none"> 1) Sistem menampilkan notifikasi pesanan sudah di-cancel <p>S-2: Buat pesanan</p> <ol style="list-style-type: none"> 1) Dapur membuat pesanan 2) Dapur mengonfirmasi pesanan sudah dibuat 3) Kasir mengonfirmasi pesanan sudah selesai 		
<i>Alternative/Exceptional Flows:</i>		

B.21 Memberikan Diskon Pesanan

Tabel B.21 *Use Case Description* Memberikan Diskon Pesanan

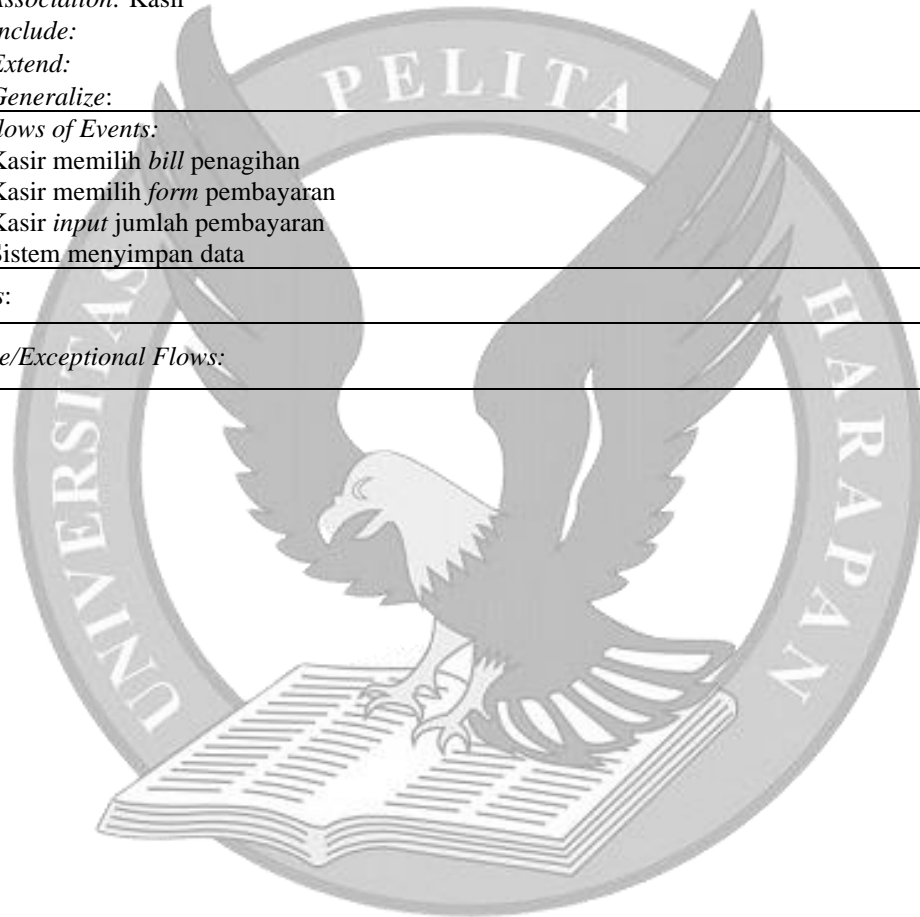
<i>Use Case Name:</i> Memberikan Diskon Pesanan	ID: 21	<i>Importance level:</i> medium
<i>Primary Actor:</i> Kasir	<i>Use Case Type:</i> Detail, essential	
<i>Stakeholders and Interests:</i> Kasir – Ingin memberikan diskon ke pesanan pelanggan		
<i>Brief Description:</i> Use case ini menjelaskan bagaimana kasir memberikan diskon ke pesanan pelanggan		
<i>Trigger:</i> Pelanggan mendapatkan diskon <i>Type:</i> External		
<i>Relationships:</i> <ul style="list-style-type: none">• <i>Association:</i> Kasir• <i>Include:</i>• <i>Extend:</i>• <i>Generalize:</i>		
<i>Normal Flows of Events:</i> <ol style="list-style-type: none">1) Kasir memilih <i>form</i> diskon2) Kasir <i>input</i> jumlah diskon3) Sistem menyimpan data		
<i>Sub Flows:</i>		
<i>Alternative/Exceptional Flows:</i>		



B.22 Bayar Pesanan

Tabel B.22 *Use Case Description* Bayar Pesanan

<i>Use Case Name:</i> Bayar Pesanan	ID: 22	<i>Importance level:</i> high
<i>Primary Actor:</i> Kasir	<i>Use Case Type:</i> Detail, essential	
<i>Stakeholders and Interests:</i> Kasir – Menerima pembayaran pelanggan Pelanggan – Ingin membayar pesanan		
<i>Brief Description:</i> <i>Use case ini menjelaskan bagaimana kasir menyimpan data pembayaran pesanan</i>		
<i>Trigger:</i> Pelanggan membayar pesanan <i>Type:</i> External		
<i>Relationships:</i> <ul style="list-style-type: none">• <i>Association:</i> Kasir• <i>Include:</i>• <i>Extend:</i>• <i>Generalize:</i>		
<i>Normal Flows of Events:</i> <ol style="list-style-type: none">1) Kasir memilih <i>bill</i> penagihan2) Kasir memilih <i>form</i> pembayaran3) Kasir <i>input</i> jumlah pembayaran4) Sistem menyimpan data		
<i>Sub Flows:</i>		
<i>Alternative/Exceptional Flows:</i>		



LAMPIRAN C
USER ACCEPTANCE TEST

SCENARIO #2: SIGN UP

Scenario Description

Skenario ini menjelaskan mengenai proses *user* melakukan pendaftaran akun ke dalam sistem

Test Scripts

Skenario ini terdiri dari *test script(s)* berikut:

- 1) *Sign Up user baru*

Use Case

- 1) *Sign Up*

User Group

- 1) Kasir
- 2) Dapur

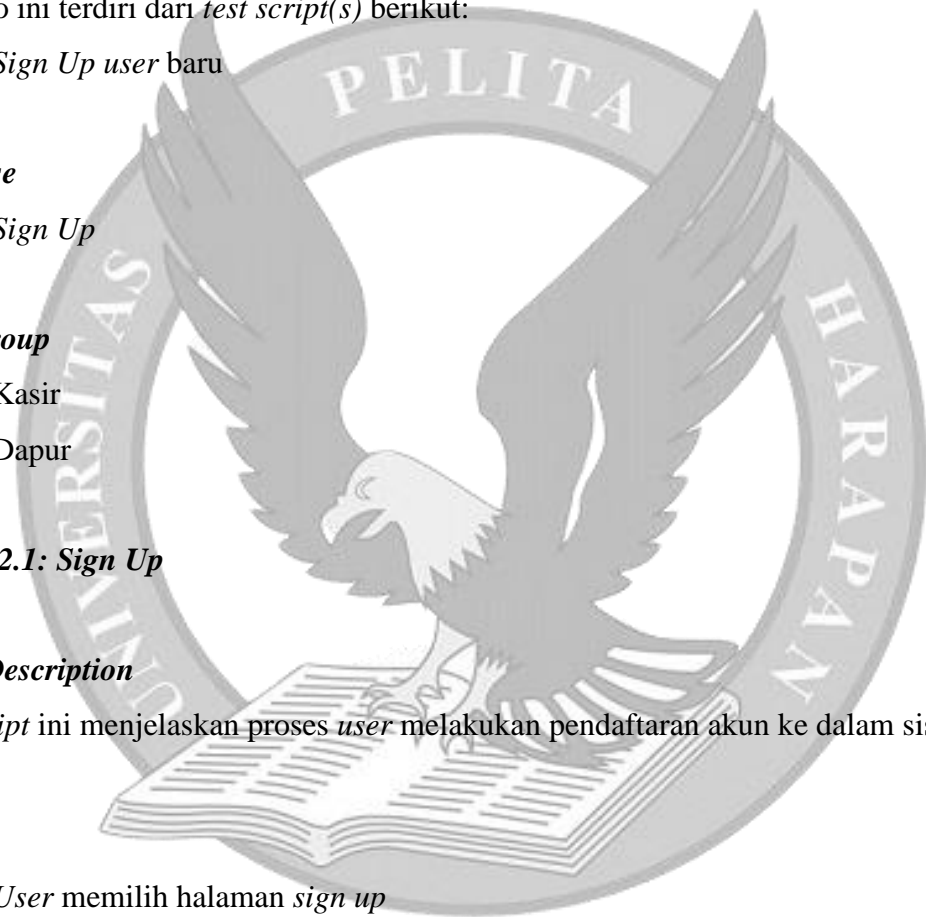
Script #2.1: Sign Up

Script Description

Test script ini menjelaskan proses *user* melakukan pendaftaran akun ke dalam sistem

Setup

- 1) *User* memilih halaman *sign up*



Script Steps

Tabel C.1 Script #2.1 Sign Up

Step #	Test Action	Expected Results	Status	Reference
1	User menekan link register dari halaman log in	Sistem menampilkan halaman registrasi	Pass	Gambar 4.83
2	User menekan tombol register tanpa mengisi data	Sistem menampilkan notifikasi pada data yang tidak diisi	Pass	Gambar 4.84
3	User mengisi dengan data yang sudah ada dan menekan tombol register	Sistem menampilkan notifikasi data sudah tercantum ke dalam database	Pass	Gambar 4.85
4	User mengisi dengan data yang belum ada dan menekan tombol register	Sistem menyimpan data baru dan menampilkan halaman log in	Pass	Gambar 4.79
5	User menekan tombol balik ke halaman log in	Sistem menampilkan halaman log in	Pass	Gambar 4.79



SCENARIO #3: MENGELOLA USER

Scenario Description

Skenario ini menjelaskan mengenai proses manajer mengelola *master data user* pada sistem

Test Scripts

Skenario ini terdiri dari *test script(s)* berikut:

- 1) Menambah data *user* baru
- 2) Mengubah data *user*
- 3) Mendeaktivasikan *user*
- 4) Memilih PDF *user*

Use Case

- 1) Mengelola *User*

User Group

- 1) Manajer

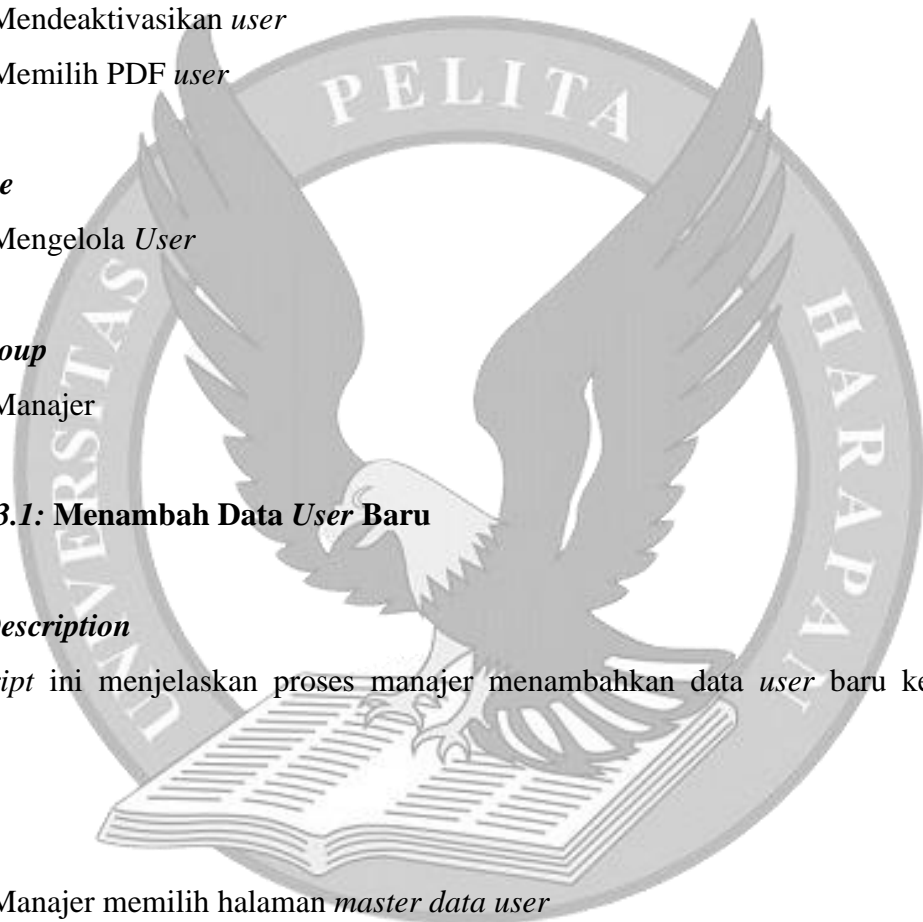
Script #3.1: Menambah Data User Baru

Script Description

Test script ini menjelaskan proses manajer menambahkan data *user* baru ke dalam sistem

Setup

- 1) Manajer memilih halaman *master data user*



Script Steps

Tabel C.2 Script #3.1 Menambah Data User Baru

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan tombol “Tambah”	Sistem menampilkan halaman <i>form</i> penambahan data baru	<i>Pass</i>	Gambar 4.90
2	Manajer menekan tombol “Submit” tanpa mengisi <i>form</i>	Sistem menampilkan notifikasi pada data yang tidak diisi	<i>Pass</i>	-
3	Manajer mengisi <i>form</i> dengan data yang sudah ada dan menekan tombol “Submit”	Sistem menampilkan notifikasi data sudah tercantum ke dalam <i>database</i>	<i>Pass</i>	Gambar 4.92
4	Manajer mengisi <i>form</i> dengan data yang belum ada dan menekan tombol “Submit”	Sistem menyimpan data baru dan menampilkan halaman <i>master data user</i>	<i>Pass</i>	Gambar 4.89
5	Manajer menekan tombol “Back”	Sistem menampilkan halaman <i>master data user</i>	<i>Pass</i>	Gambar 4.89

Script #3.2: Mengubah Data User

Script Description

Test script ini menjelaskan proses manajer mengubah data *user* pada sistem

Setup

- 1) Manajer memilih halaman *master data user*

Script Steps

Tabel C.3 Script #3.2 Mengubah Data User

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan tombol "Edit"	Sistem menampilkan halaman <i>form</i> perubahan data	<i>Pass</i>	Gambar 4.91
2	Manajer mengosongkan <i>form</i> dan menekan tombol "Submit"	Sistem menampilkan notifikasi pada data yang tidak diisi	<i>Pass</i>	-
3	Manajer mengubah data pada <i>form</i> dengan data yang sudah ada dan menekan tombol "Submit"	Sistem menampilkan notifikasi data sudah tercantum ke dalam <i>database</i>	<i>Pass</i>	Gambar 4.92
4	Manajer mengubah data pada <i>form</i> dengan data yang belum ada dan menekan tombol "Submit"	Sistem menyimpan data baru dan menampilkan halaman <i>master data user</i>	<i>Pass</i>	Gambar 4.89
5	Manajer menekan tombol "Back"	Sistem menampilkan halaman <i>master data user</i>	<i>Pass</i>	Gambar 4.89

Script #3.3: Mendeaktivasikan User

Script Description

Test script ini menjelaskan proses manajer mendeaktivasikan *user*

Setup

- 1) Manajer memilih halaman *master data user*

Script Steps

Tabel C.4 Script #3.3 Mendeaktivasikan User

Step #	Test Action	Expected Results	Status	Reference
1	Manajer menekan tombol “ <i>Deactivate</i> ”	Sistem menampilkan notifikasi konfirmasi deaktivasi data	<i>Pass</i>	Gambar 4.93
2	Manajer menekan tombol “OK” pada notifikasi konfirmasi	Sistem mengubah status data menjadi “ <i>inactive</i> ” dan menampilkan halaman <i>master data user</i>	<i>Pass</i>	Gambar 4.89

Script #3.4: Memilih PDF User

Script Description

Test script ini menjelaskan proses manajer memilih halaman PDF data user

Setup

- 1) Manajer memilih halaman *master data user*

Script Steps

Tabel C.5 Script #3.4 Memilih PDF User

Step #	Test Action	Expected Results	Status	Reference
1	Manajer menekan tombol “ <i>PDF</i> ”	Sistem menampilkan halaman PDF user	<i>Pass</i>	Gambar 4.94

SCENARIO #4: MENGELOLA ROLE

Scenario Description

Skenario ini menjelaskan mengenai proses manajer mengelola *master data role* pada sistem

Test Scripts

Skenario ini terdiri dari *test script(s)* berikut:

- 1) Menambah data *role* baru
- 2) Mengubah data *role*
- 3) Mendeaktivasikan *role*
- 4) Memilih PDF *role*

Use Case

- 1) Mengelola *Role*

User Group

- 1) Manajer

Script #4.1: Menambah Data Role Baru

Script Description

Test script ini menjelaskan proses manajer menambahkan data *role* baru ke dalam sistem

Setup

- 1) Manajer memilih halaman *master data role*



Script Steps

Tabel C.6 Script #4.1 Menambah Data Role Baru

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan tombol “Tambah”	Sistem menampilkan halaman <i>form</i> penambahan data baru	<i>Pass</i>	Gambar 4.96
2	Manajer menekan tombol “Submit” tanpa mengisi <i>form</i>	Sistem menampilkan notifikasi pada data yang tidak diisi	<i>Pass</i>	-
3	Manajer mengisi <i>form</i> dengan data yang sudah ada dan menekan tombol “Submit”	Sistem menampilkan notifikasi data sudah tercantum ke dalam <i>database</i>	<i>Pass</i>	Gambar 4.98
4	Manajer mengisi <i>form</i> dengan data yang belum ada dan menekan tombol “Submit”	Sistem menyimpan data baru dan menampilkan halaman <i>master data role</i>	<i>Pass</i>	Gambar 4.95
5	Manajer menekan tombol “Back”	Sistem menampilkan halaman <i>master data role</i>	<i>Pass</i>	Gambar 4.95

Script #4.2: Mengubah Data Role

Script Description

Test script ini menjelaskan proses manajer mengubah data *role* pada sistem

Setup

- 1) Manajer memilih halaman *master data role*

Script Steps

Tabel C.7 Script #4.2 Mengubah Data Role

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan tombol "Edit"	Sistem menampilkan halaman <i>form</i> perubahan data	<i>Pass</i>	Gambar 4.97
2	Manajer mengosongkan <i>form</i> dan menekan tombol "Submit"	Sistem menampilkan notifikasi pada data yang tidak diisi	<i>Pass</i>	-
3	Manajer mengubah data pada <i>form</i> dengan data yang sudah ada dan menekan tombol "Submit"	Sistem menampilkan notifikasi data sudah tercantum ke dalam <i>database</i>	<i>Pass</i>	Gambar 4.98
4	Manajer mengubah data pada <i>form</i> dengan data yang belum ada dan menekan tombol "Submit"	Sistem menyimpan data baru dan menampilkan halaman <i>master data role</i>	<i>Pass</i>	Gambar 4.95
5	Manajer menekan tombol "Back"	Sistem menampilkan halaman <i>master data role</i>	<i>Pass</i>	Gambar 4.95

Script #4.3: Mendeaktivasikan Role

Script Description

Test script ini menjelaskan proses manajer mendeaktivasikan *role*

Setup

- 1) Manajer memilih halaman *master data role*

Script Steps

Tabel C.8 Script #4.3 Mendeaktivasikan Role

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan tombol “ <i>Deactivate</i> ”	Sistem menampilkan notifikasi konfirmasi deaktivasi data	<i>Pass</i>	Gambar 4.99
2	Manajer menekan tombol “OK” pada notifikasi konfirmasi	Sistem mengubah status data menjadi “ <i>inactive</i> ” dan menampilkan halaman <i>master data role</i>	<i>Pass</i>	Gambar 4.95

Script #4.4: Memilih PDF Role

Script Description

Test script ini menjelaskan proses manajer memilih halaman PDF data *role*

Setup

- 1) Manajer memilih halaman *master data role*

Script Steps

Tabel C.9 Script #4.4 Memilih PDF Role

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan tombol “ <i>PDF</i> ”	Sistem menampilkan halaman PDF <i>role</i>	<i>Pass</i>	Gambar 4.100

SCENARIO #5: MENGELOLA MENU

Scenario Description

Skenario ini menjelaskan mengenai proses manajer mengelola *master data menu* pada sistem

Test Scripts

Skenario ini terdiri dari *test script(s)* berikut:

- 1) Menambah data *menu* baru
- 2) Mengubah data *menu*
- 3) Mendeaktivasikan *menu*
- 4) Mengaktivasikan *menu*
- 5) Memilih PDF *menu*

Use Case

- 1) Mengelola *Menu Makanan*

User Group

- 1) Manajer

Script #5.1: Menambah Data Menu Baru

Script Description

Test script ini menjelaskan proses manajer menambahkan data *menu* baru ke dalam sistem

Setup

- 1) Manajer memilih halaman *master data menu*



Script Steps

Tabel C.10 Script #5.1 Menambah Data Menu Baru

Step #	Test Action	Expected Results	Status	Reference
1	Manajer menekan tombol “Tambah”	Sistem menampilkan halaman <i>form</i> penambahan data baru	<i>Pass</i>	Gambar 4.102
2	Manajer menekan tombol “Submit” tanpa mengisi <i>form</i>	Sistem menampilkan notifikasi pada data yang tidak diisi	<i>Pass</i>	-
3	Manajer mengisi <i>form</i> dengan data yang sudah ada dan menekan tombol “Submit”	Sistem menampilkan notifikasi data sudah tercantum ke dalam <i>database</i>	<i>Pass</i>	Gambar 4.105
4	Manajer mengisi <i>form</i> dengan data yang belum ada dan menekan tombol “Submit”	Sistem menyimpan data baru dan menampilkan halaman <i>master data menu</i>	<i>Pass</i>	Gambar 4.101
5	Manajer menekan tombol “Back”	Sistem menampilkan halaman <i>master data menu</i>	<i>Pass</i>	Gambar 4.101

Script #5.2: Mengubah Data Menu

Script Description

Test script ini menjelaskan proses manajer mengubah data *menu* pada sistem

Setup

- 1) Manajer memilih halaman *master data menu*

Script Steps

Tabel C.11 Script #5.2 Mengubaha Data Menu

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan tombol "Edit"	Sistem menampilkan halaman <i>form</i> perubahan data	<i>Pass</i>	Gambar 4.103
2	Manajer mengosongkan <i>form</i> dan menekan tombol "Submit"	Sistem menampilkan notifikasi pada data yang tidak diisi	<i>Pass</i>	-
3	Manajer mengubah data pada <i>form</i> dengan data yang sudah ada dan menekan tombol "Submit"	Sistem menampilkan notifikasi data sudah tercantum ke dalam <i>database</i>	<i>Pass</i>	Gambar 4.105
4	Manajer mengubah data pada <i>form</i> dengan data yang belum ada dan menekan tombol "Submit"	Sistem menyimpan data baru dan menampilkan halaman <i>master data menu</i>	<i>Pass</i>	Gambar 4.101
5	Manajer menekan tombol "Back"	Sistem menampilkan halaman <i>master data menu</i>	<i>Pass</i>	Gambar 4.101

Script #5.3: Mendeaktivasikan Menu

Script Description

Test script ini menjelaskan proses manajer mendeaktivasikan *menu*

Setup

- 1) Manajer memilih halaman *master data menu*

Script Steps

Tabel C.12 *Script #5.3 Mendeaktivasikan Menu*

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan tombol “ <i>Deactivate</i> ”	Sistem mengubah status data menjadi “ <i>inactive</i> ”	<i>Pass</i>	Gambar 4.106

Script #5.4: Mengaktivasikan Menu

Script Description

Test script ini menjelaskan proses manajer mengaktivasikan *menu*

Setup

- 1) Manajer memilih halaman *master data menu*

Script Steps

Tabel C.13 *Script #5.4 Mengaktivasikan Menu*

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan tombol “ <i>Activate</i> ”	Sistem mengubah status data menjadi “ <i>active</i> ”	<i>Pass</i>	Gambar 4.106

Script #5.5: Memilih PDF Menu

Script Description

Test script ini menjelaskan proses manajer memilih halaman PDF data *menu*

Setup

- 1) Manajer memilih halaman *master data menu*

Script Steps

Tabel C.14 *Script #5.5 Memilih PDF Menu*

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan tombol “ <i>PDF</i> ”	Sistem menampilkan halaman PDF <i>menu</i>	<i>Pass</i>	Gambar 4.107

SCENARIO #6: MENGELOLA DETAIL MENU

Scenario Description

Skenario ini menjelaskan mengenai proses manajer mengelola *master data detail menu* pada sistem

Test Scripts

Skenario ini terdiri dari *test script(s)* berikut:

- 1) Memilih data *detail menu*
- 2) Menambah data *detail menu* baru
- 3) Mengubah data *detail menu*
- 4) Menghapus *detail menu*

Use Case

- 1) Mengelola *Menu Detail*

User Group

- 1) Manajer

Script #6.1: Memilih Data Detail Menu

Script Description

Test script ini menjelaskan proses manajer memilih halaman data *detail menu*

Setup

- 1) Manajer memilih halaman *master data menu*



Script Steps

Tabel C.15 Script #6.1 Memilih Data Detail Menu

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan tombol “Detail” pada halaman <i>master data menu</i>	Sistem menampilkan halaman data <i>detail menu</i>	<i>Pass</i>	Gambar 4.108

Script #6.2: Menambah Data Detail Menu Baru

Script Description

Test script ini menjelaskan proses manajer menambahkan data *detail menu* baru ke dalam sistem

Setup

- 1) Manajer memilih halaman data *detail menu*

Script Steps

Tabel C.16 Script #6.2 Menambah Data Detail Menu

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan tombol “Tambah”	Sistem menampilkan halaman <i>form</i> penambahan data baru	<i>Pass</i>	Gambar 4.109
2	Manajer menekan tombol “Submit” tanpa mengisi <i>form</i>	Sistem menampilkan notifikasi pada data yang tidak diisi	<i>Pass</i>	-
3	Manajer mengisi <i>form</i> dengan data yang sudah ada dan menekan tombol “Submit”	Sistem menampilkan notifikasi data sudah tercantum ke dalam <i>database</i>	<i>Pass</i>	Gambar 4.111
4	Manajer mengisi <i>form</i> dengan data yang belum ada dan menekan tombol “Submit”	Sistem menyimpan data baru dan menampilkan halaman data <i>detail menu</i>	<i>Pass</i>	Gambar 4.108
5	Manajer menekan tombol “Back”	Sistem menampilkan halaman data <i>detail menu</i>	<i>Pass</i>	Gambar 4.108

Script #6.3: Mengubah Data Detail Menu

Script Description

Test script ini menjelaskan proses manajer mengubah data *detail menu* pada sistem

Setup

- 1) Manajer memilih halaman data *detail menu*

Script Steps

Tabel C.17 *Script #6.3 Mengubah Data Detail Menu*

Step #	Test Action	Expected Results	Status	Reference
1	Manajer menekan tombol "Edit"	Sistem menampilkan halaman <i>form</i> perubahan data	<i>Pass</i>	Gambar 4.110
2	Manajer mengosongkan <i>form</i> dan menekan tombol "Submit"	Sistem menampilkan notifikasi pada data yang tidak diisi	<i>Pass</i>	-
3	Manajer mengubah data pada <i>form</i> dengan data yang sudah ada dan menekan tombol "Submit"	Sistem menampilkan notifikasi data sudah tercantum ke dalam <i>database</i>	<i>Pass</i>	Gambar 4.111
4	Manajer mengubah data pada <i>form</i> dengan data yang belum ada dan menekan tombol "Submit"	Sistem menyimpan data baru dan menampilkan halaman data <i>detail menu</i>	<i>Pass</i>	Gambar 4.108
5	Manajer menekan tombol "Back"	Sistem menampilkan halaman data <i>detail menu</i>	<i>Pass</i>	Gambar 4.108

Script #6.4: Menghapus Menu Detail

Script Description

Test script ini menjelaskan proses manajer mendeaktivasikan *detail menu*

Setup

- 1) Manajer memilih halaman data *detail menu*

Script Steps

Tabel C.18 Script #6.4 Menghapus Detail Menu

Step #	Test Action	Expected Results	Status	Reference
1	Manajer menekan tombol "Delete"	Sistem menampilkan notifikasi konfirmasi hapus data	Pass	Gambar 4.112
2	Manajer menekan tombol "OK" pada notifikasi konfirmasi	Sistem menghapus data pada sistem dan menampilkan halaman data <i>detail menu</i>	Pass	Gambar 4.108



SCENARIO #7: MENGELOLA KATEGORI MENU

Scenario Description

Skenario ini menjelaskan mengenai proses manajer mengelola *master data* kategori *menu* pada sistem

Test Scripts

Skenario ini terdiri dari *test script(s)* berikut:

- 1) Menambah data kategori baru
- 2) Mengubah data kategori
- 3) Mendeaktivasikan kategori
- 4) Memilih PDF kategori

Use Case

- 1) Mengelola Kategori *Menu*

User Group

- 1) Manajer

Script #7.1: Menambah Data Kategori Baru

Script Description

Test script ini menjelaskan proses manajer menambahkan data kategori baru ke dalam sistem

Setup

- 1) Manajer memilih halaman *master data* kategori



Script Steps

Tabel C.19 Script #7.1 Menambah Data Kategori Baru

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan tombol “Tambah”	Sistem menampilkan halaman <i>form</i> penambahan data baru	<i>Pass</i>	Gambar 4.114
2	Manajer menekan tombol “Submit” tanpa mengisi <i>form</i>	Sistem menampilkan notifikasi pada data yang tidak diisi	<i>Pass</i>	-
3	Manajer mengisi <i>form</i> dengan data yang sudah ada dan menekan tombol “Submit”	Sistem menampilkan notifikasi data sudah tercantum ke dalam <i>database</i>	<i>Pass</i>	Gambar 4.116
4	Manajer mengisi <i>form</i> dengan data yang belum ada dan menekan tombol “Submit”	Sistem menyimpan data baru dan menampilkan halaman <i>master data</i> kategori	<i>Pass</i>	Gambar 4.113
5	Manajer menekan tombol “Back”	Sistem menampilkan halaman <i>master data</i> kategori	<i>Pass</i>	Gambar 4.113

Script #7.2: Mengubah Data Kategori

Script Description

Test script ini menjelaskan proses manajer mengubah data kategori pada sistem

Setup

- 1) Manajer memilih halaman *master data* kategori

Script Steps

Tabel C.20 Script #7.2 Mengubah Data Kategori

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan tombol "Edit"	Sistem menampilkan halaman <i>form</i> perubahan data	<i>Pass</i>	Gambar 4.115
2	Manajer mengosongkan <i>form</i> dan menekan tombol "Submit"	Sistem menampilkan notifikasi pada data yang tidak diisi	<i>Pass</i>	-
3	Manajer mengubah data pada <i>form</i> dengan data yang sudah ada dan menekan tombol "Submit"	Sistem menampilkan notifikasi data sudah tercantum ke dalam <i>database</i>	<i>Pass</i>	Gambar 4.116
4	Manajer mengubah data pada <i>form</i> dengan data yang belum ada dan menekan tombol "Submit"	Sistem menyimpan data baru dan menampilkan halaman <i>master data</i> kategori	<i>Pass</i>	Gambar 4.113
5	Manajer menekan tombol "Back"	Sistem menampilkan halaman <i>master data</i> kategori	<i>Pass</i>	Gambar 4.113

Script #7.3: Mendeaktivasikan Kategori

Script Description

Test script ini menjelaskan proses manajer mendeaktivasikan kategori

Setup

- 1) Manajer memilih halaman *master data* kategori

Script Steps

Tabel C.21 Script #7.3 Mendeaktivasikan Kategori

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan tombol “ <i>Deactivate</i> ”	Sistem menampilkan notifikasi konfirmasi deaktivasi data	<i>Pass</i>	Gambar 4.117
2	Manajer menekan tombol “OK” pada notifikasi konfirmasi	Sistem mengubah status data menjadi “ <i>inactive</i> ” dan menampilkan halaman <i>master data</i> kategori	<i>Pass</i>	Gambar 4.113

Script #7.4: Memilih PDF Kategori

Script Description

Test script ini menjelaskan proses manajer memilih halaman PDF data kategori

Setup

- 1) Manajer memilih halaman *master data* kategori

Script Steps

Tabel C.22 Script #7.4 Memilih PDF Kategori

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan tombol “ <i>PDF</i> ”	Sistem menampilkan halaman PDF kategori	<i>Pass</i>	Gambar 4.118

SCENARIO #8: MENGELOLA INVENTORI BARANG

Scenario Description

Skenario ini menjelaskan mengenai proses manajer mengelola *master data* inventori barang pada sistem

Test Scripts

Skenario ini terdiri dari *test script(s)* berikut:

- 1) Menambah data inventori baru
- 2) Mengubah data inventori
- 3) Menambah jumlah stok barang
- 4) Mendeaktivasikan inventori
- 5) Memilih PDF inventori

Use Case

- 1) Mengelola Inventori Barang
- 2) Menambah Stok Inventori Barang

User Group

- 1) Manajer

Script #8.1: Menambah Data Inventori Baru

Script Description

Test script ini menjelaskan proses manajer menambahkan data inventori baru ke dalam sistem

Setup

- 1) Manajer memilih halaman *master data* inventori



Script Steps

Tabel C.23 Script #8.1 Menambah Data Inventori Baru

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan tombol “Tambah”	Sistem menampilkan halaman <i>form</i> penambahan data baru	<i>Pass</i>	Gambar 4.120
2	Manajer menekan tombol “Submit” tanpa mengisi <i>form</i>	Sistem menampilkan notifikasi pada data yang tidak diisi	<i>Pass</i>	-
3	Manajer mengisi <i>form</i> dengan data yang sudah ada dan menekan tombol “Submit”	Sistem menampilkan notifikasi data sudah tercantum ke dalam <i>database</i>	<i>Pass</i>	Gambar 4.123
4	Manajer mengisi <i>form</i> dengan data yang belum ada dan menekan tombol “Submit”	Sistem menyimpan data baru dan menampilkan halaman <i>master data</i> inventori	<i>Pass</i>	Gambar 4.119
5	Manajer menekan tombol “Back”	Sistem menampilkan halaman <i>master data</i> inventori	<i>Pass</i>	Gambar 4.119

Script #8.2: Mengubah Data Inventori

Script Description

Test script ini menjelaskan proses manajer mengubah data inventori pada sistem

Setup

- 1) Manajer memilih halaman *master data* inventori

Script Steps

Tabel C.24 Script #8.2 Mengubah Data Inventori

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan tombol "Edit"	Sistem menampilkan halaman <i>form</i> perubahan data	<i>Pass</i>	Gambar 4.121
2	Manajer mengosongkan <i>form</i> dan menekan tombol "Submit"	Sistem menampilkan notifikasi pada data yang tidak diisi	<i>Pass</i>	-
3	Manajer mengubah data pada <i>form</i> dengan data yang sudah ada dan menekan tombol "Submit"	Sistem menampilkan notifikasi data sudah tercantum ke dalam <i>database</i>	<i>Pass</i>	Gambar 4.123
4	Manajer mengubah data pada <i>form</i> dengan data yang belum ada dan menekan tombol "Submit"	Sistem menyimpan data baru dan menampilkan halaman <i>master data</i> inventori	<i>Pass</i>	Gambar 4.119
5	Manajer menekan tombol "Back"	Sistem menampilkan halaman <i>master data</i> inventori	<i>Pass</i>	Gambar 4.119

Script #8.3: Menambah Jumlah Stok Barang

Script Description

Test script ini menjelaskan proses manajer menambahkan jumlah stok barang pada sistem

Setup

- 1) Manajer memilih halaman *master data* inventori

Script Steps

Tabel C.25 Script #8.3 Menambah Jumlah Stok Barang

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan tombol "Add"	Sistem menampilkan halaman penambahan stok barang	Pass	Gambar 4.122
2	Manajer mengisi jumlah penambahan stok dan menekan tombol "Submit"	Sistem mengubah data stok barang pada sistem dan menampilkan halaman <i>master data</i> inventori	Pass	Gambar 4.119
3	Manajer menekan tombol "Back"	Sistem menampilkan halaman <i>master data</i> inventori	Pass	Gambar 4.119

Script #8.4: Mendeaktivasikan Inventori

Script Description

Test script ini menjelaskan proses manajer mendeaktivasikan inventori

Setup

- 1) Manajer memilih halaman *master data* inventori

Script Steps

Tabel C.26 Script #8.4 Mendeaktivasikan Inventori

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan tombol "Deactivate"	Sistem menampilkan notifikasi konfirmasi deaktivasi data	Pass	Gambar 4.124

Script #8.5: Mengaktivasikan Inventori

Script Description

Test script ini menjelaskan proses manajer mengaktivasikan inventori

Setup

- 1) Manajer memilih halaman *master data* inventori

Script Steps

Tabel C.27 Script #8.5 Mengaktivasikan Inventori

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
---------------	--------------------	-------------------------	---------------	------------------

Tabel C.27 Script #8.5 Mengaktifasikan Inventori

1	Manajer menekan tombol "Activate"	Sistem mengubah status data menjadi "active"	Pass	Gambar 4.124
---	-----------------------------------	--	------	--------------

Script #8.6: Memilih PDF Inventori

Script Description

Test script ini menjelaskan proses manajer memilih halaman PDF data inventori

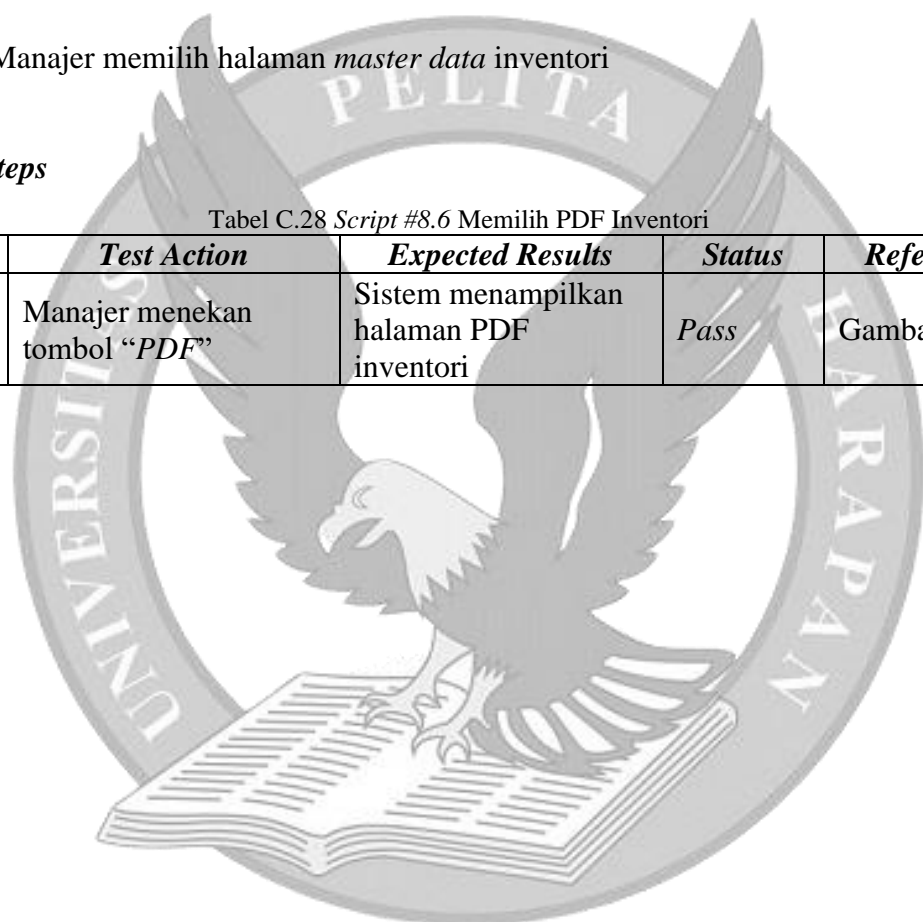
Setup

- 1) Manajer memilih halaman *master data* inventori

Script Steps

Tabel C.28 Script #8.6 Memilih PDF Inventori

Step #	Test Action	Expected Results	Status	Reference
1	Manajer menekan tombol "PDF"	Sistem menampilkan halaman PDF inventori	Pass	Gambar 4.125



SCENARIO #9: MENGELOLA JENIS BARANG

Scenario Description

Skenario ini menjelaskan mengenai proses manajer mengelola *master data* jenis barang pada sistem

Test Scripts

Skenario ini terdiri dari *test script(s)* berikut:

- 1) Menambah data jenis barang baru
- 2) Mengubah data jenis barang
- 3) Mendeaktivasikan jenis barang
- 4) Memilih PDF jenis barang

Use Case

- 1) Mengelola Jenis Barang

User Group

- 1) Manajer

Script #9.1: Menambah Data Jenis Barang Baru

Script Description

Test script ini menjelaskan proses manajer menambahkan data jenis barang baru ke dalam sistem

Setup

- 1) Manajer memilih halaman *master data* jenis barang



Script Steps

Tabel C.29 Script #9.1 Menambah Data Jenis Barang Baru

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan tombol “Tambah”	Sistem menampilkan halaman <i>form</i> penambahan data baru	<i>Pass</i>	Gambar 4.127
2	Manajer menekan tombol “Submit” tanpa mengisi <i>form</i>	Sistem menampilkan notifikasi pada data yang tidak diisi	<i>Pass</i>	-
3	Manajer mengisi <i>form</i> dengan data yang sudah ada dan menekan tombol “Submit”	Sistem menampilkan notifikasi data sudah tercantum ke dalam <i>database</i>	<i>Pass</i>	Gambar 4.129
4	Manajer mengisi <i>form</i> dengan data yang belum ada dan menekan tombol “Submit”	Sistem menyimpan data baru dan menampilkan halaman <i>master data</i> jenis barang	<i>Pass</i>	Gambar 4.126
5	Manajer menekan tombol “Back”	Sistem menampilkan halaman <i>master data</i> jenis barang	<i>Pass</i>	Gambar 4.126

Script #9.2: Mengubah Data Jenis Barang

Script Description

Test script ini menjelaskan proses manajer mengubah data jenis barang pada sistem

Setup

- 1) Manajer memilih halaman *master data* jenis barang

Script Steps

Tabel C.30 Script #9.2 Mengubah Data Jenis Barang

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan tombol "Edit"	Sistem menampilkan halaman <i>form</i> perubahan data	<i>Pass</i>	Gambar 4.128
2	Manajer mengosongkan <i>form</i> dan menekan tombol "Submit"	Sistem menampilkan notifikasi pada data yang tidak diisi	<i>Pass</i>	-
3	Manajer mengubah data pada <i>form</i> dengan data yang sudah ada dan menekan tombol "Submit"	Sistem menampilkan notifikasi data sudah tercantum ke dalam <i>database</i>	<i>Pass</i>	Gambar 4.129
4	Manajer mengubah data pada <i>form</i> dengan data yang belum ada dan menekan tombol "Submit"	Sistem menyimpan data baru dan menampilkan halaman <i>master data</i> jenis barang	<i>Pass</i>	Gambar 4.126
5	Manajer menekan tombol "Back"	Sistem menampilkan halaman <i>master data</i> jenis barang	<i>Pass</i>	Gambar 4.126

Script #9.3: Mendeaktivasikan Jenis Barang

Script Description

Test script ini menjelaskan proses manajer mendeaktivasikan jenis barang

Setup

- 1) Manajer memilih halaman *master data* jenis barang

Script Steps

Tabel C.31 Script #9.3 Mendeaktivasikan Jenis Barang

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan tombol “ <i>Deactivate</i> ”	Sistem menampilkan notifikasi konfirmasi deaktivasi data	<i>Pass</i>	Gambar 4.130
2	Manajer menekan tombol “OK” pada notifikasi konfirmasi	Sistem mengubah status data menjadi “ <i>inactive</i> ” dan menampilkan halaman <i>master data</i> jenis barang	<i>Pass</i>	Gambar 4.126

Script #9.4: Memilih PDF Jenis Barang

Script Description

Test script ini menjelaskan proses manajer memilih halaman PDF data jenis barang

Setup

- 1) Manajer memilih halaman *master data* jenis barang

Script Steps

Tabel C.32 Script #9.4 Memilih PDF Jenis Barang

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan tombol “ <i>PDF</i> ”	Sistem menampilkan halaman PDF jenis barang	<i>Pass</i>	Gambar 4.131

SCENARIO #10: MENGELOLA SATUAN BARANG

Scenario Description

Skenario ini menjelaskan mengenai proses manajer mengelola *master data* satuan barang pada sistem

Test Scripts

Skenario ini terdiri dari *test script(s)* berikut:

- 1) Menambah data satuan baru
- 2) Mengubah data satuan
- 3) Mendeaktivasikan satuan
- 4) Memilih PDF satuan

Use Case

- 1) Mengelola Satuan Barang

User Group

- 1) Manajer

Script #10.1: Menambah Data Barang Baru

Script Description

Test script ini menjelaskan proses manajer menambahkan data satuan baru ke dalam sistem

Setup

- 1) Manajer memilih halaman *master data* satuan



Script Steps

Tabel C.33 Script #10.1 Menambah Data Satuan Baru

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan tombol “Tambah”	Sistem menampilkan halaman <i>form</i> penambahan data baru	<i>Pass</i>	Gambar 4.133
2	Manajer menekan tombol “Submit” tanpa mengisi <i>form</i>	Sistem menampilkan notifikasi pada data yang tidak diisi	<i>Pass</i>	-
3	Manajer mengisi <i>form</i> dengan data yang sudah ada dan menekan tombol “Submit”	Sistem menampilkan notifikasi data sudah tercantum ke dalam <i>database</i>	<i>Pass</i>	Gambar 4.135
4	Manajer mengisi <i>form</i> dengan data yang belum ada dan menekan tombol “Submit”	Sistem menyimpan data baru dan menampilkan halaman <i>master data</i> satuan	<i>Pass</i>	Gambar 4.132
5	Manajer menekan tombol “Back”	Sistem menampilkan halaman <i>master data</i> satuan	<i>Pass</i>	Gambar 4.132

Script #10.2: Mengubah Data Satuan

Script Description

Test script ini menjelaskan proses manajer mengubah data satuan pada sistem

Setup

- 1) Manajer memilih halaman *master data* satuan

Script Steps

Tabel C.34 Script #10.2 Mengubah Data Satuan

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan tombol "Edit"	Sistem menampilkan halaman <i>form</i> perubahan data	<i>Pass</i>	Gambar 4.134
2	Manajer mengosongkan <i>form</i> dan menekan tombol "Submit"	Sistem menampilkan notifikasi pada data yang tidak diisi	<i>Pass</i>	-
3	Manajer mengubah data pada <i>form</i> dengan data yang sudah ada dan menekan tombol "Submit"	Sistem menampilkan notifikasi data sudah tercantum ke dalam <i>database</i>	<i>Pass</i>	Gambar 4.135
4	Manajer mengubah data pada <i>form</i> dengan data yang belum ada dan menekan tombol "Submit"	Sistem menyimpan data baru dan menampilkan halaman <i>master data</i> satuan	<i>Pass</i>	Gambar 4.132
5	Manajer menekan tombol "Back"	Sistem menampilkan halaman <i>master data</i> satuan	<i>Pass</i>	Gambar 4.132

Script #10.3: Mendeaktivasikan Satuan

Script Description

Test script ini menjelaskan proses manajer mendeaktivasikan satuan

Setup

- 1) Manajer memilih halaman *master data* satuan

Script Steps

Tabel C.35 Script #10.3 Mendeaktivasikan Satuan

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan tombol "Deactivate"	Sistem menampilkan notifikasi konfirmasi deaktivasi data	<i>Pass</i>	Gambar 4.136
2	Manajer menekan tombol "OK" pada notifikasi konfirmasi	Sistem mengubah status data menjadi "inactive" dan menampilkan halaman <i>master data</i> satuan	<i>Pass</i>	Gambar 4.132

Script #10.4: Memilih PDF Satuan

Script Description

Test script ini menjelaskan proses manajer memilih halaman PDF data satuan

Setup

- 1) Manajer memilih halaman *master data* satuan

Script Steps

Tabel C.36 *Script #10.4* Memilih PDF Satuan

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan tombol “PDF”	Sistem menampilkan halaman PDF satuan	<i>Pass</i>	Gambar 4.137



SCENARIO #11: MENGELOLA PEMASUKAN

Scenario Description

Skenario ini menjelaskan mengenai proses manajer mengelola *master data* pemasukan pada sistem

Test Scripts

Skenario ini terdiri dari *test script(s)* berikut:

- 1) *Sorting* data pemasukan
- 2) Memilih data *detail* pesanan
- 3) Memilih PDF pemasukan

Use Case

- 1) Mengelola Pemasukan

User Group

- 1) Manajer

Script #11.1: Sorting Data Pemasukan

Script Description

Test script ini menjelaskan proses manajer melakukan *sorting* data pemasukan pada sistem

Setup

- 1) Manajer memilih halaman *master data* pemasukan

Script Steps

Tabel C.37 *Script #11.1* Sorting Data Pemasukan

Step #	Test Action	Expected Results	Status	Reference
1	Manajer mengisi tanggal awal dan akhir, lalu menekan tombol “Sort”	Sistem menampilkan data pemasukan hasil <i>sorting</i>	<i>Pass</i>	Gambar 4.138

Script #11.2: Memilih Data Pesanan Detail

Script Description

Test script ini menjelaskan proses manajer memilih data *detail* pesanan

Setup

- 1) Manajer memilih halaman *master data* pesanan

Script Steps

Tabel C.38 *Script #11.2* Memilih Data *Detail* Pesanan

Step #	Test Action	Expected Results	Status	Reference
1	Manajer menekan tombol " <i>Detail</i> "	Sistem menampilkan halaman data <i>detail</i> pesanan	<i>Pass</i>	Gambar 4.140

Script #11.3: Memilih PDF Pemasukan

Script Description

Test script ini menjelaskan proses manajer memilih halaman PDF data pemasukan

Setup

- 1) Manajer memilih halaman *master data* pemasukan

Script Steps

Tabel C.39 *Script #11.3* Memilih PDF Pemasukan

Step #	Test Action	Expected Results	Status	Reference
1	Manajer menekan tombol " <i>PDF</i> "	Sistem menampilkan halaman PDF pemasukan	<i>Pass</i>	Gambar 4.139

SCENARIO #12: MENGELOLA MEJA

Scenario Description

Skenario ini menjelaskan mengenai proses manajer mengelola *master data* meja pada sistem

Test Scripts

Skenario ini terdiri dari *test script(s)* berikut:

- 1) Menambah data meja baru
- 2) Mengubah data meja
- 3) Mendeaktivasikan meja
- 4) Memilih PDF meja

Use Case

- 1) Mengelola Meja

User Group

- 1) Manajer

Script #12.1: Menambah Data Meja Baru

Script Description

Test script ini menjelaskan proses manajer menambahkan data meja baru ke dalam sistem

Setup

- 1) Manajer memilih halaman *master data* meja



Script Steps

Tabel C.40 Script #12.1 Menambah Data Meja Baru

Step #	Test Action	Expected Results	Status	Reference
1	Manajer menekan tombol “Tambah”	Sistem menampilkan halaman <i>form</i> penambahan data baru	<i>Pass</i>	Gambar 4.142
2	Manajer menekan tombol “Submit” tanpa mengisi <i>form</i>	Sistem menampilkan notifikasi pada data yang tidak diisi	<i>Pass</i>	-
3	Manajer mengisi <i>form</i> dengan data yang sudah ada dan menekan tombol “Submit”	Sistem menampilkan notifikasi data sudah tercantum ke dalam <i>database</i>	<i>Pass</i>	Gambar 4.144
4	Manajer mengisi <i>form</i> dengan data yang belum ada dan menekan tombol “Submit”	Sistem menyimpan data baru dan menampilkan halaman <i>master data</i> meja	<i>Pass</i>	Gambar 4.141
5	Manajer menekan tombol “Back”	Sistem menampilkan halaman <i>master data</i> meja	<i>Pass</i>	Gambar 4.141

Script #12.2: Mengubah Data Meja

Script Description

Test script ini menjelaskan proses manajer mengubah data meja pada sistem

Setup

- 1) Manajer memilih halaman *master data* meja

Script Steps

Tabel C.41 Script #12.2 Mengubah Data Meja

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan tombol "Edit"	Sistem menampilkan halaman <i>form</i> perubahan data	<i>Pass</i>	Gambar 4.143
2	Manajer mengosongkan <i>form</i> dan menekan tombol "Submit"	Sistem menampilkan notifikasi pada data yang tidak diisi	<i>Pass</i>	-
3	Manajer mengubah data pada <i>form</i> dengan data yang sudah ada dan menekan tombol "Submit"	Sistem menampilkan notifikasi data sudah tercantum ke dalam <i>database</i>	<i>Pass</i>	Gambar 4.144
4	Manajer mengubah data pada <i>form</i> dengan data yang belum ada dan menekan tombol "Submit"	Sistem menyimpan data baru dan menampilkan halaman <i>master data</i> meja	<i>Pass</i>	Gambar 4.141
5	Manajer menekan tombol "Back"	Sistem menampilkan halaman <i>master data</i> meja	<i>Pass</i>	Gambar 4.141

Script #12.3: Mendeaktivasikan Meja

Script Description

Test script ini menjelaskan proses manajer mendeaktivasikan meja

Setup

- 1) Manajer memilih halaman *master data* meja

Script Steps

Tabel C.42 Script #12.3 Mendeaktivasikan Meja

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan tombol “ <i>Deactivate</i> ”	Sistem menampilkan notifikasi konfirmasi deaktivasi data	<i>Pass</i>	Gambar 4.145
2	Manajer menekan tombol “OK” pada notifikasi konfirmasi	Sistem mengubah status data menjadi “ <i>inactive</i> ” dan menampilkan halaman <i>master data</i> meja	<i>Pass</i>	Gambar 4.141

Script #12.4: Memilih PDF Meja

Script Description

Test script ini menjelaskan proses manajer memilih halaman PDF data meja

Setup

- 1) Manajer memilih halaman *master data* meja

Script Steps

Tabel C.43 Script #12.4 Memilih PDF Meja

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan tombol “ <i>PDF</i> ”	Sistem menampilkan halaman PDF meja	<i>Pass</i>	Gambar 4.146

SCENARIO #13: MENGELOLA METODE PEMBAYARAN

Scenario Description

Skenario ini menjelaskan mengenai proses manajer mengelola *master data* metode bayar pada sistem

Test Scripts

Skenario ini terdiri dari *test script(s)* berikut:

- 1) Menambah data metode bayar baru
- 2) Mengubah data metode bayar
- 3) Mendeaktivasikan metode bayar
- 4) Memilih PDF metode bayar

Use Case

- 1) Mengelola Metode Pembayaran

User Group

- 1) Manajer

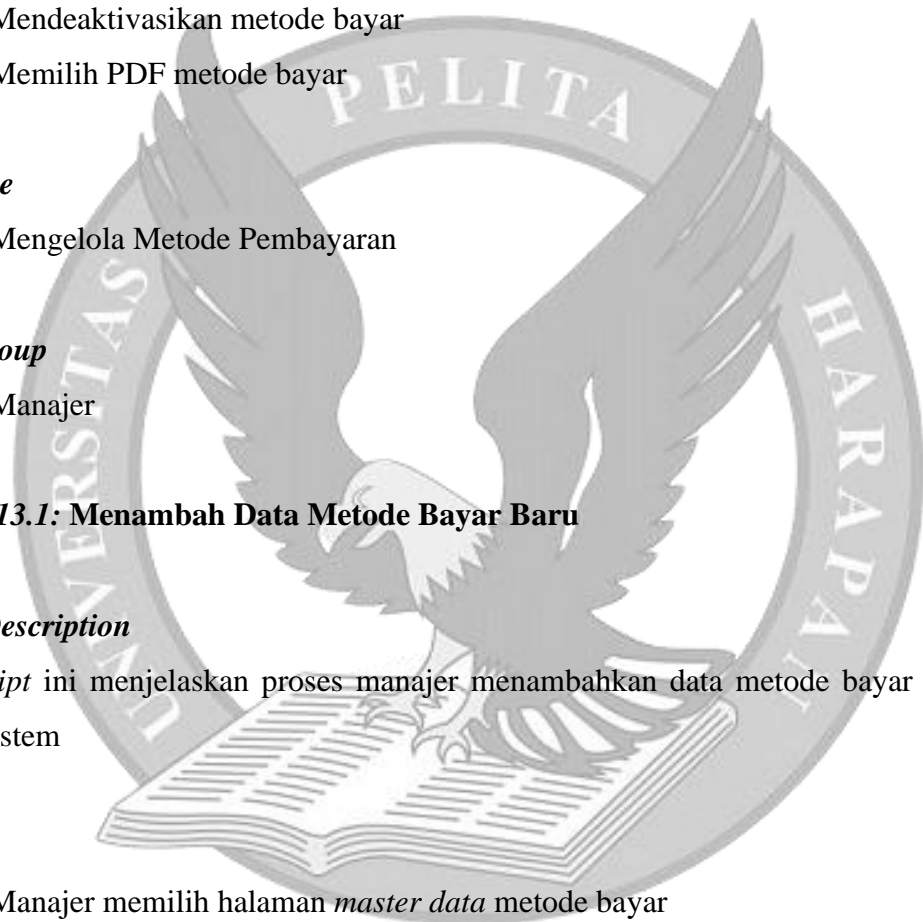
Script #13.1: Menambah Data Metode Bayar Baru

Script Description

Test script ini menjelaskan proses manajer menambahkan data metode bayar baru ke dalam sistem

Setup

- 1) Manajer memilih halaman *master data* metode bayar



Script Steps

Tabel C.44 Script #13.1 Menambah Data Metode Bayar Baru

Step #	Test Action	Expected Results	Status	Reference
1	Manajer menekan tombol “Tambah”	Sistem menampilkan halaman <i>form</i> penambahan data baru	<i>Pass</i>	Gambar 4.148
2	Manajer menekan tombol “Submit” tanpa mengisi <i>form</i>	Sistem menampilkan notifikasi pada data yang tidak diisi	<i>Pass</i>	-
3	Manajer mengisi <i>form</i> dengan data yang sudah ada dan menekan tombol “Submit”	Sistem menampilkan notifikasi data sudah tercantum ke dalam <i>database</i>	<i>Pass</i>	Gambar 4.150
4	Manajer mengisi <i>form</i> dengan data yang belum ada dan menekan tombol “Submit”	Sistem menyimpan data baru dan menampilkan halaman <i>master data</i> metode bayar	<i>Pass</i>	Gambar 4.147
5	Manajer menekan tombol “Back”	Sistem menampilkan halaman <i>master data</i> metode bayar	<i>Pass</i>	Gambar 4.147

Script #13.2: Mengubah Data Metode Bayar

Script Description

Test script ini menjelaskan proses manajer mengubah data metode bayar pada sistem

Setup

- 1) Manajer memilih halaman *master data* metode bayar

Script Steps

Tabel C.45 Script #13.2 Mengubah Data Metode Bayar

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan tombol "Edit"	Sistem menampilkan halaman <i>form</i> perubahan data	<i>Pass</i>	Gambar 4.149
2	Manajer mengosongkan <i>form</i> dan menekan tombol "Submit"	Sistem menampilkan notifikasi pada data yang tidak diisi	<i>Pass</i>	-
3	Manajer mengubah data pada <i>form</i> dengan data yang sudah ada dan menekan tombol "Submit"	Sistem menampilkan notifikasi data sudah tercantum ke dalam <i>database</i>	<i>Pass</i>	Gambar 4.150
4	Manajer mengubah data pada <i>form</i> dengan data yang belum ada dan menekan tombol "Submit"	Sistem menyimpan data baru dan menampilkan halaman <i>master data</i> metode bayar	<i>Pass</i>	Gambar 4.147
5	Manajer menekan tombol "Back"	Sistem menampilkan halaman <i>master data</i> metode bayar	<i>Pass</i>	Gambar 4.147

Script #13.3: Mendeaktivasikan Metode Bayar

Script Description

Test script ini menjelaskan proses manajer mendeaktivasikan metode bayar

Setup

- 1) Manajer memilih halaman *master data* metode bayar

Script Steps

Tabel C.46 Script #13.3 Mendeaktivasikan Metode Bayar

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan tombol “ <i>Deactivate</i> ”	Sistem menampilkan notifikasi konfirmasi deaktivasi data	<i>Pass</i>	Gambar 4.151
2	Manajer menekan tombol “OK” pada notifikasi konfirmasi	Sistem mengubah status data menjadi “ <i>inactive</i> ” dan menampilkan halaman <i>master data</i> metode bayar	<i>Pass</i>	Gambar 4.147

Script #13.4: Memilih PDF Metode Bayar

Script Description

Test script ini menjelaskan proses manajer memilih halaman PDF data metode bayar

Setup

- 1) Manajer memilih halaman *master data* metode bayar

Script Steps

Tabel C.47 Script #13.4 Memilih PDF Metode Bayar

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan tombol “ <i>PDF</i> ”	Sistem menampilkan halaman PDF metode bayar	<i>Pass</i>	Gambar 4.152

SCENARIO #14: MENGELOLA PELANGGAN

Scenario Description

Skenario ini menjelaskan mengenai proses manajer mengelola *master data* pelanggan pada sistem

Test Scripts

Skenario ini terdiri dari *test script(s)* berikut:

- 1) Menambah data pelanggan baru
- 2) Mengubah data pelanggan
- 3) Mendeaktivasikan pelanggan
- 4) Memilih PDF pelanggan

Use Case

- 1) Mengelola Pelanggan

User Group

- 1) Manajer

Script #14.1: Menambah Data Pelanggan Baru

Script Description

Test script ini menjelaskan proses manajer menambahkan data pelanggan baru ke dalam sistem

Setup

- 1) Manajer memilih halaman *master data* pelanggan



Script Steps

Tabel C.48 Script #14.1 Menambah Data Pelanggan Baru

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan tombol “Tambah”	Sistem menampilkan halaman <i>form</i> penambahan data baru	<i>Pass</i>	Gambar 4.154
2	Manajer menekan tombol “ <i>Submit</i> ” tanpa mengisi <i>form</i>	Sistem menampilkan notifikasi pada data yang tidak diisi	<i>Pass</i>	-
3	Manajer mengisi <i>form</i> dengan data yang sudah ada dan menekan tombol “ <i>Submit</i> ”	Sistem menampilkan notifikasi data sudah tercantum ke dalam <i>database</i>	<i>Pass</i>	Gambar 4.156
4	Manajer mengisi <i>form</i> dengan data yang belum ada dan menekan tombol “ <i>Submit</i> ”	Sistem menyimpan data baru dan menampilkan halaman <i>master data</i> pelanggan	<i>Pass</i>	Gambar 4.153
5	Manajer menekan tombol “Back”	Sistem menampilkan halaman <i>master data</i> pelanggan	<i>Pass</i>	Gambar 4.153

Script #14.2: Mengubah Data Pelanggan

Script Description

Test script ini menjelaskan proses manajer mengubah data pelanggan pada sistem

Setup

- 1) Manajer memilih halaman *master data* pelanggan

Script Steps

Tabel C.49 Script #14.2 Mengubah Data Pelanggan

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan tombol "Edit"	Sistem menampilkan halaman <i>form</i> perubahan data	<i>Pass</i>	Gambar 4.155
2	Manajer mengosongkan <i>form</i> dan menekan tombol "Submit"	Sistem menampilkan notifikasi pada data yang tidak diisi	<i>Pass</i>	-
3	Manajer mengubah data pada <i>form</i> dengan data yang sudah ada dan menekan tombol "Submit"	Sistem menampilkan notifikasi data sudah tercantum ke dalam <i>database</i>	<i>Pass</i>	Gambar 4.156
4	Manajer mengubah data pada <i>form</i> dengan data yang belum ada dan menekan tombol "Submit"	Sistem menyimpan data baru dan menampilkan halaman <i>master data</i> pelanggan	<i>Pass</i>	Gambar 4.153
5	Manajer menekan tombol "Back"	Sistem menampilkan halaman <i>master data</i> pelanggan	<i>Pass</i>	Gambar 4.153

Script #14.3: Mendeaktivasikan Pelanggan

Script Description

Test script ini menjelaskan proses manajer mendeaktivasikan pelanggan

Setup

- 1) Manajer memilih halaman *master data* pelanggan

Script Steps

Tabel C.50 Script #14.3 Mendeaktivasikan Pelanggan

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan tombol "Deactivate"	Sistem menampilkan notifikasi konfirmasi deaktivasi data	<i>Pass</i>	Gambar 4.157
2	Manajer menekan tombol "OK" pada notifikasi konfirmasi	Sistem mengubah status data menjadi "inactive" dan menampilkan halaman <i>master data</i> pelanggan	<i>Pass</i>	Gambar 4.153

Script #14.4: Memilih PDF Pelanggan

Script Description

Test script ini menjelaskan proses manajer memilih halaman PDF data pelanggan

Setup

- 1) Manajer memilih halaman *master data* pelanggan

Script Steps

Tabel C.51 *Script #14.4* Memilih PDF Pelanggan

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan tombol “PDF”	Sistem menampilkan halaman PDF pelanggan	<i>Pass</i>	Gambar 4.158



SCENARIO #15: MENGELOLA PENGELUARAN

Scenario Description

Skenario ini menjelaskan mengenai proses manajer mengelola *master data* pengeluaran pada sistem

Test Scripts

Skenario ini terdiri dari *test script(s)* berikut:

- 1) *Sorting* data pengeluaran
- 2) Menambah data pengeluaran baru
- 3) Mengubah data pengeluaran
- 4) Memilih PDF pengeluaran

Use Case

- 1) Mengelola Pengeluaran

User Group

- 1) Manajer

Script #15.1: Sorting Data Pengeluaran

Script Description

Test script ini menjelaskan proses manajer melakukan *sorting* data pengeluaran pada sistem

Setup

- 1) Manajer memilih halaman *master data* pengeluaran

Script Steps

Tabel C.52 *Script #15.1 Sorting Data Pengeluaran*

Step #	Test Action	Expected Results	Status	Reference
1	Manajer mengisi tanggal awal dan akhir, lalu menekan tombol “Sort”	Sistem menampilkan data pengeluaran hasil <i>sorting</i>	<i>Pass</i>	Gambar 4.159

Script #15.2: Menambah Data Pengeluaran Baru

Script Description

Test script ini menjelaskan proses manajer menambahkan data pengeluaran baru ke dalam sistem

Setup

- 1) Manajer memilih halaman *master data* pengeluaran

Script Steps

Tabel C.53 *Script #15.2* Menambah Data Pengeluaran Baru

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan tombol “Tambah”	Sistem menampilkan halaman <i>form</i> penambahan data baru	<i>Pass</i>	Gambar 4.160
2	Manajer menekan tombol “ <i>Submit</i> ” tanpa mengisi <i>form</i>	Sistem menampilkan notifikasi pada data yang tidak diisi	<i>Pass</i>	-
3	Manajer mengisi <i>form</i> dengan data yang sudah ada dan menekan tombol “ <i>Submit</i> ”	Sistem menampilkan notifikasi data sudah tercantum ke dalam <i>database</i>	<i>Pass</i>	Gambar 4.162
4	Manajer mengisi <i>form</i> dengan data yang belum ada dan menekan tombol “ <i>Submit</i> ”	Sistem menyimpan data baru dan menampilkan halaman <i>master data</i> pengeluaran	<i>Pass</i>	Gambar 4.159
5	Manajer menekan tombol “Back”	Sistem menampilkan halaman <i>master data</i> pengeluaran	<i>Pass</i>	Gambar 4.159

Script #15.3: Mengubah Data Pengeluaran

Script Description

Test script ini menjelaskan proses manajer mengubah data pengeluaran pada sistem

Setup

- 1) Manajer memilih halaman *master data* pengeluaran

Script Steps

Tabel C.54 *Script #15.3* Mengubah Data Pengeluaran

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan	Sistem menampilkan	<i>Pass</i>	Gambar 4.161

Tabel C.54 Script #15.3 Mengubah Data Pengeluaran

	tombol "Edit"	halaman <i>form</i> perubahan data		
2	Manajer mengosongkan <i>form</i> dan menekan tombol "Submit"	Sistem menampilkan notifikasi pada data yang tidak diisi	Pass	-
3	Manajer mengubah data pada <i>form</i> dengan data yang sudah ada dan menekan tombol "Submit"	Sistem menampilkan notifikasi data sudah tercantum ke dalam <i>database</i>	Pass	Gambar 4.162
4	Manajer mengubah data pada <i>form</i> dengan data yang belum ada dan menekan tombol "Submit"	Sistem menyimpan data baru dan menampilkan halaman <i>master data</i> pengeluaran	Pass	Gambar 4.159
5	Manajer menekan tombol "Back"	Sistem menampilkan halaman <i>master data</i> pengeluaran	Pass	Gambar 4.159

Script #15.4: Memilih PDF Pengeluaran

Script Description

Test script ini menjelaskan proses manajer memilih halaman PDF data pengeluaran

Setup

- 1) Manajer memilih halaman *master data* pengeluaran

Script Steps

Tabel C.55 Script #15.4 Memilih PDF Pengeluaran

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan tombol "PDF"	Sistem menampilkan halaman PDF pengeluaran	Pass	Gambar 4.163

SCENARIO #16: MENGELOLA SUPPLIER

Scenario Description

Skenario ini menjelaskan mengenai proses manajer mengelola *master data supplier* pada sistem

Test Scripts

Skenario ini terdiri dari *test script(s)* berikut:

- 1) Menambah data *supplier* baru
- 2) Mengubah data *supplier*
- 3) Mendeaktivasikan *supplier*
- 4) Memilih PDF *supplier*

Use Case

- 1) Mengelola *Supplier*

User Group

- 1) Manajer

Script #16.1: Menambah Data Supplier Baru

Script Description

Test script ini menjelaskan proses manajer menambahkan data *supplier* baru ke dalam sistem

Setup

- 1) Manajer memilih halaman *master data supplier*



Script Steps

Tabel C.56 Script #16.1 Menambah Data Supplier

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan tombol “Tambah”	Sistem menampilkan halaman <i>form</i> penambahan data baru	<i>Pass</i>	Gambar 4.165
2	Manajer menekan tombol “Submit” tanpa mengisi <i>form</i>	Sistem menampilkan notifikasi pada data yang tidak diisi	<i>Pass</i>	-
3	Manajer mengisi <i>form</i> dengan data yang sudah ada dan menekan tombol “Submit”	Sistem menampilkan notifikasi data sudah tercantum ke dalam <i>database</i>	<i>Pass</i>	Gambar 4.167
4	Manajer mengisi <i>form</i> dengan data yang belum ada dan menekan tombol “Submit”	Sistem menyimpan data baru dan menampilkan halaman <i>master data supplier</i>	<i>Pass</i>	Gambar 4.164
5	Manajer menekan tombol “Back”	Sistem menampilkan halaman <i>master data supplier</i>	<i>Pass</i>	Gambar 4.164

Script #16.2: Mengubah Data Supplier

Script Description

Test script ini menjelaskan proses manajer mengubah data *supplier* pada sistem

Setup

- 1) Manajer memilih halaman *master data supplier*

Script Steps

Tabel C.57 Script #16.2 Mengubah Data Supplier

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan tombol "Edit"	Sistem menampilkan halaman <i>form</i> perubahan data	<i>Pass</i>	Gambar 4.166
2	Manajer mengosongkan <i>form</i> dan menekan tombol "Submit"	Sistem menampilkan notifikasi pada data yang tidak diisi	<i>Pass</i>	-
3	Manajer mengubah data pada <i>form</i> dengan data yang sudah ada dan menekan tombol "Submit"	Sistem menampilkan notifikasi data sudah tercantum ke dalam <i>database</i>	<i>Pass</i>	Gambar 4.167
4	Manajer mengubah data pada <i>form</i> dengan data yang belum ada dan menekan tombol "Submit"	Sistem menyimpan data baru dan menampilkan halaman <i>master data supplier</i>	<i>Pass</i>	Gambar 4.164
5	Manajer menekan tombol "Back"	Sistem menampilkan halaman <i>master data supplier</i>	<i>Pass</i>	Gambar 4.164

Script #16.3: Mendeaktivasikan Supplier

Script Description

Test script ini menjelaskan proses manajer mendeaktivasikan *supplier*

Setup

- 1) Manajer memilih halaman *master data supplier*

Script Steps

Tabel C.58 Script #16.3 Mendeaktivasikan Supplier

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan tombol “ <i>Deactivate</i> ”	Sistem menampilkan notifikasi konfirmasi deaktivasi data	<i>Pass</i>	Gambar 4.168
2	Manajer menekan tombol “OK” pada notifikasi konfirmasi	Sistem mengubah status data menjadi “ <i>inactive</i> ” dan menampilkan halaman <i>master data supplier</i>	<i>Pass</i>	Gambar 4.164

Script #16.4: Memilih PDF Supplier

Script Description

Test script ini menjelaskan proses manajer memilih halaman PDF data *supplier*

Setup

- 1) Manajer memilih halaman *master data supplier*

Script Steps

Tabel C.59 Script #16.4 Memilih PDF Supplier

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Manajer menekan tombol “ <i>PDF</i> ”	Sistem menampilkan halaman PDF <i>supplier</i>	<i>Pass</i>	Gambar 4.169

SCENARIO #17: MENAMBAH PESANAN PELANGGAN

Scenario Description

Skenario ini menjelaskan mengenai proses kasir menambahkan pesanan pelanggan ke dalam sistem

Test Scripts

Skenario ini terdiri dari *test script(s)* berikut:

- 1) Menambah pesanan

Use Case

- 1) Mengelola Pesanan Pelanggan

User Group

- 1) Kasir

Script #17.1: Mengelola Pesanan Pelanggan

Script Description

Test script ini menjelaskan proses menambah pesanan ke dalam sistem

Setup

- 1) Kasir memilih halaman *dashboard* kasir



Script Steps

Tabel C.60 Script #17.1 Mengelola Pesanan Pelanggan

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Kasir menekan tombol "Tambah Pesanan"	Sistem menampilkan halaman <i>form</i> penambahan data pesanan	<i>Pass</i>	Gambar 4.175
2	Kasir mengisi data dan menekan tombol "Submit"	Sistem menyimpan data baru dan menampilkan halaman <i>dashboard</i> kasir	<i>Pass</i>	Gambar 4.170
3	Kasir menekan tombol "Back"	Sistem menampilkan halaman <i>dashboard</i> kasir	<i>Pass</i>	Gambar 4.170



SCENARIO #18: MENGELOLA *DETAIL* PESANAN PELANGGAN

Scenario Description

Skenario ini menjelaskan mengenai proses kasir mengelola *detail* pesanan pelanggan

Test Scripts

Skenario ini terdiri dari *test script(s)* berikut:

- 1) Menambah *menu* ke dalam pesanan
- 2) Memberikan diskon
- 3) Membatalkan *menu* pada pesanan
- 4) Menyajikan *menu* pada pesanan

Use Case

- 1) Mengelola *Detail* Pesanan
- 2) Memberikan Diskon Pesanan

User Group

- 1) Kasir

Script #18.1: Menambah Menu Ke Dalam Pesanan

Script Description

Test script ini menjelaskan proses menambah *menu* ke dalam pesanan

Setup

- 1) Kasir memilih halaman *detail* pesanan



Script Steps

Tabel C.61 Script #18.1 Menambah Menu Ke Dalam Pesanan

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Kasir menekan tombol “Tambah” dengan status pesanan belum selesai	Sistem menampilkan halaman <i>form</i> penambahan <i>menu</i> ke dalam pesanan	<i>Pass</i>	Gambar 4.179
2	Kasir menekan tombol “Tambah” dengan status pesanan sudah selesai	Sistem menampilkan notifikasi pesanan sudah selesai	<i>Pass</i>	Gambar 4.180
3	Kasir menekan tombol “Submit” tanpa mengisi <i>form</i>	Sistem menampilkan notifikasi pada data yang tidak diisi	<i>Pass</i>	-
4	Kasir mengisi data <i>menu</i> yang memiliki stok bahan yang kurang dan menekan tombol “Submit”	Sistem menampilkan notifikasi stok bahan kurang	<i>Pass</i>	Gambar 4.181
5	Kasir mengisi data <i>menu</i> yang memiliki stok bahan yang cukup dan menekan tombol “Submit”	Sistem menyimpan data baru dan menampilkan halaman <i>detail</i> pesanan	<i>Pass</i>	Gambar 4.176
6	Kasir menekan tombol “Back”	Sistem menampilkan halaman <i>detail</i> pesanan	<i>Pass</i>	Gambar 4.176

Script #18..2: Memberikan Diskon

Script Description

Test script ini menjelaskan proses pemberian diskon kepada *menu* pada pesanan

Setup

- 1) Kasir memilih halaman *detail* pesanan

Script Steps

Tabel C.62 Script #18.2 Memberikan Diskon

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Kasir menekan tombol "Diskon" dengan status pesanan belum selesai	Sistem menampilkan halaman <i>form</i> pemberian diskon	<i>Pass</i>	Gambar 4.182
2	Kasir menekan tombol "Diskon" dengan status pesanan sudah selesai	Sistem menampilkan notifikasi pesan sudah selesai	<i>Pass</i>	Gambar 4.180
3	Kasir mengisi jumlah diskon	Sistem menyimpan data baru dan menampilkan halaman <i>detail</i> pesanan	<i>Pass</i>	Gambar 4.176
4	Kasir menekan tombol "Back"	Sistem menampilkan halaman <i>detail</i> pesanan	<i>Pass</i>	Gambar 4.176

Script #18.3: Membatalkan Menu Pada Pesanan

Script Description

Test script ini menjelaskan proses pembatalan *menu* pada pesanan

Setup

- 1) Kasir memilih halaman *detail* pesanan

Script Steps

Tabel C.63 Script #18.3 Membatalkan Menu Pada Pesanan

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Kasir menekan tombol "Cancel" dengan status penyajian "belum diterima"	Sistem mengubah status penyajian menjadi "cancelled"	<i>Pass</i>	Gambar 4.177
2	Kasir menekan tombol "Cancel" dengan status penyajian "lagi dimasak", "sudah selesai", "sudah disajikan"	Sistem menampilkan notifikasi pesan sudah diterima	<i>Pass</i>	Gambar 4.178

Script #18.4: Menyajikan Menu Pada Pesanan

Script Description

Test script ini menjelaskan proses penyajian *menu* pada pesanan

Setup

- 1) Kasir memilih halaman *detail* pesanan

Script Steps

Tabel C.64 Script #18.4 Menyajikan Menu Pada Pesanan

Step #	Test Action	Expected Results	Status	Reference
1	Kasir menekan tombol “Selesai” dengan status penyajian “belum diterima”	Sistem menampilkan notifikasi pesanan belum diterima	<i>Pass</i>	-
2	Kasir menekan tombol “Selesai” dengan status penyajian “lagi dimasak”	Sistem menampilkan notifikasi pesanan lagi dimasak	<i>Pass</i>	Gambar 4.178
3	Kasir menekan tombol “Selesai” dengan status penyajian “sudah selesai”	Sistem mengubah status penyajian menjadi “sudah disajikan”	<i>Pass</i>	Gambar 4.177



SCENARIO #19: MELAKUKAN KONFIRMASI PESANAN

Scenario Description

Skenario ini menjelaskan mengenai proses dapur melakukan konfirmasi pesanan baru

Test Scripts

Skenario ini terdiri dari *test script(s)* berikut:

- 1) Menerima pesanan baru
- 2) Menyelesaikan pesanan

Use Case

- 1) Melakukan Konfirmasi Pesanan

User Group

- 1) Dapur

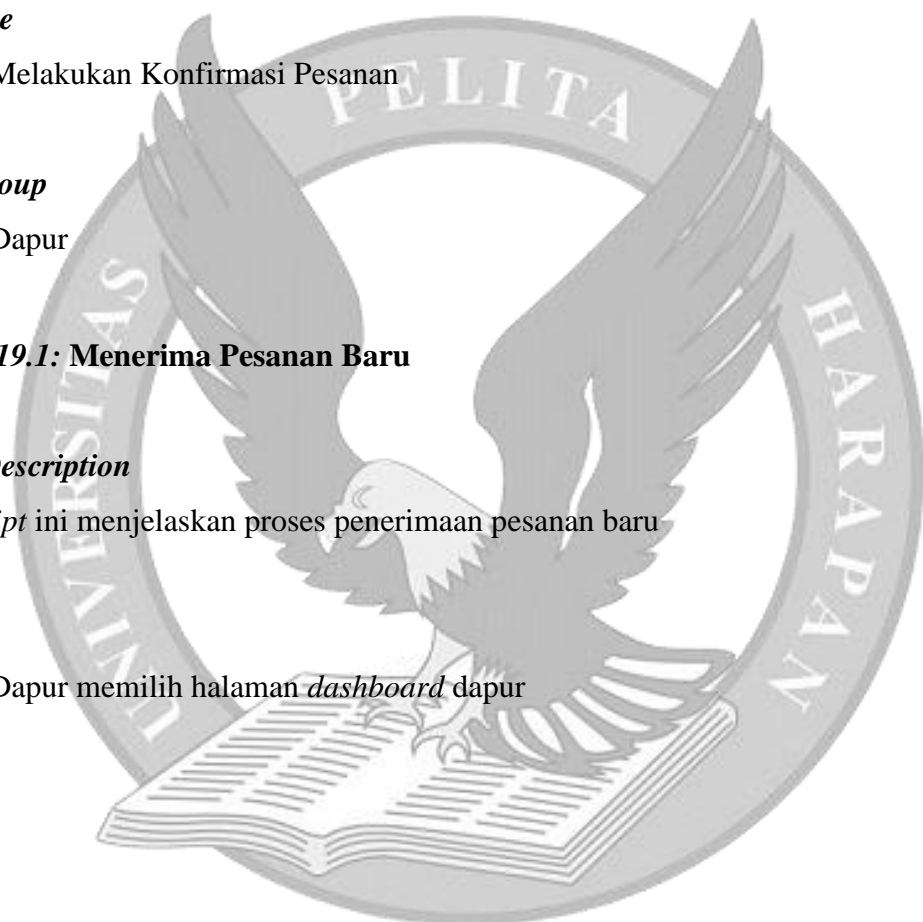
Script #19.1: Menerima Pesanan Baru

Script Description

Test script ini menjelaskan proses penerimaan pesanan baru

Setup

- 1) Dapur memilih halaman *dashboard* dapur



Script Steps

Tabel C.65 Script #19.1 Menerima Pesanan Baru

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Dapur menekan tombol “Terima” dengan status penyajian “belum diterima”	Sistem mengubah status penyajian menjadi “lagi dimasak”	<i>Pass</i>	Gambar 4.190
2	Dapur menekan tombol “Terima” dengan status penyajian “lagi dimasak”, “sudah selesai”, “sudah disajikan”	Sistem menampilkan notifikasi pesanan sudah diterima	<i>Pass</i>	Gambar 4.192
3	Dapur menekan tombol “Terima” dengan status penyajian “cancelled”	Sistem menampilkan notifikasi pesanan sudah dibatalkan	<i>Pass</i>	-

Script #19.2: Menyelesaikan Pesanan

Script Description

Test script ini menjelaskan proses penyelesaian pesanan

Setup

- 1) Dapur memilih halaman *dashboard* dapur

Script Steps

Tabel C.66 Script #19.2 Menyelesaikan Pesanan

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Dapur menekan tombol “Selesai” dengan status penyajian “belum diterima”	Sistem menampilkan notifikasi pesanan sudah diterima	<i>Pass</i>	Gambar 4.192
2	Dapur menekan tombol “Selesai” dengan status penyajian “lagi dimasak”	Sistem mengubah status penyajian menjadi “sudah selesai”	<i>Pass</i>	-
3	Dapur menekan tombol “Selesai” dengan status penyajian “cancelled”	Sistem menampilkan notifikasi pesanan sudah dibatalkan	<i>Pass</i>	-

SCENARIO #20: MELAKUKAN PEMBAYARAN

Scenario Description

Skenario ini menjelaskan mengenai proses dapur melakukan pembayaran pada pesanan

Test Scripts

Skenario ini terdiri dari *test script(s)* berikut:

- 1) Memilih *bill* penagihan
- 2) Melakukan pembayaran
- 3) Memilih *bill* pelunasan

Use Case

- 1) Bayar Pesanan

User Group

- 1) Kasir

Script #20.1: Memilih Bill Penagihan

Script Description

Test script ini menjelaskan proses menampilkan halaman *bill* penagihan

Setup

- 1) Semua *menu* pada pesanan sudah disajikan



Script Steps

Tabel C.67 Script #20.1 Memilih Bill Penagihan

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Kasir menekan tombol “Bayar” pada <i>dashboard</i> kasir	Sistem menampilkan halaman <i>bill penagihan</i>	<i>Pass</i>	Gambar 4.183
2	Kasir menekan tombol “Print” pada halaman <i>bill</i> penagihan	Sistem menampilkan halaman PDF <i>bill</i> penagihan	<i>Pass</i>	
3	Kasir menekan tombol “Bayar” pada halaman <i>billi</i> penagihan	Sistem menampilkan halaman <i>form</i> pembayaran	<i>Pass</i>	Gambar 4.185
4	Kasir menekan tombol “Home” pada halaman <i>bill</i> penagihan	Sistem menampilkan halaman <i>dashboard</i> kasir	<i>Pass</i>	Gambar 4.170

Script #20.2: Melakukan Pembayaran

Script Description

Test script ini menjelaskan proses melakukan pembayaran pesanan

Setup

- 1) Kasir memilih halaman *bill* penagihan

Script Steps

Tabel C.68 Script #20.2 Melakukan Pembayaran

<i>Step #</i>	<i>Test Action</i>	<i>Expected Results</i>	<i>Status</i>	<i>Reference</i>
1	Kasir menekan tombol “Submit” tanpa mengisi <i>form</i>	Sistem menampilkan notifikasi pada data yang tidak diisi	<i>Pass</i>	-
2	Kasir mengisi jumlah pembayaran kurang dari <i>subtotal</i> pesanan	Sistem menampilkan notifikasi pembayaran tidak cukup	<i>Pass</i>	Gambar 4.186
3	Kasir mengisi jumlah pembayaran sama dengan atau lebih dari <i>subtotal</i> pesanan	Sistem menyimpan data pembayaran dan menampilkan halaman <i>bill</i> pelunasan	<i>Pass</i>	Gambar 4.184

Script #20.3: Memilih Bill Pelunasan

Script Description

Test script ini menjelaskan proses menampilkan halaman *bill* pelunasan

Setup

- 1) Pembayaran sudah dilakukan

Script Steps

Tabel C.69 Script #20.3 Memilih Bill Pelunasan

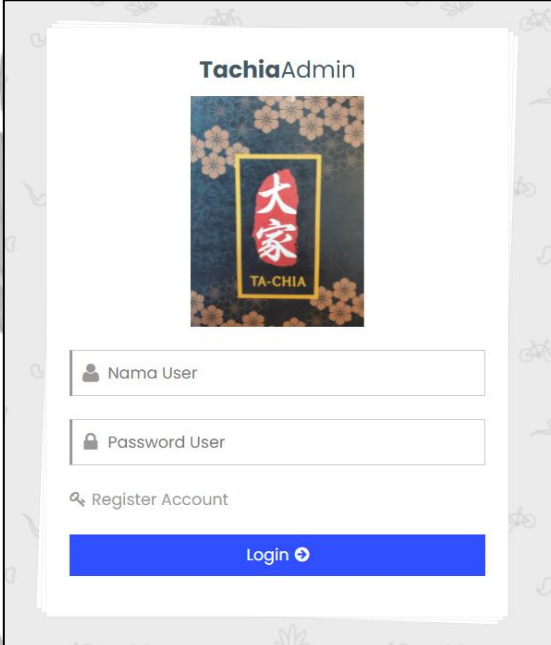
Step #	Test Action	Expected Results	Status	Reference
1	Kasir menekan tombol "Print" pada halaman <i>bill</i> pelunasan	Sistem menampilkan halaman PDF <i>bill</i> pelunasan	<i>Pass</i>	-
2	Kasir menekan tombol "Home" pada halaman <i>bill</i> pelunasan	Sistem menampilkan halaman <i>dashboard</i> kasir	<i>Pass</i>	Gambar 4.170
3	Kasir menekan tombol "Bill" pada halaman <i>dashboard</i> kasir	Sistem menampilkan halaman <i>bill</i> pelunasan	<i>Pass</i>	Gambar 4.184



LAMPIRAN D MANUAL BOOK

D.1 *Log In*

Sistem pencatatan inventori dan keuangan “Restoran Ta Chia” memiliki 3 pengguna yaitu, manajer, kasir, dan dapur. Tiap pengguna memiliki masing-masing akses kepada halaman yang berbeda. Gambar berikut merupakan halaman awal yang akan ditampilkan pertama kali untuk semua pengguna.



Gambar D.1 Halaman *Log In*

Halaman tersebut merupakan halaman untuk melakukan *log in*. Setelah mengisi field yang tersedia, tekan tombol *login* untuk mengakses halaman *dashboard* sesuai dengan *role* pengguna.

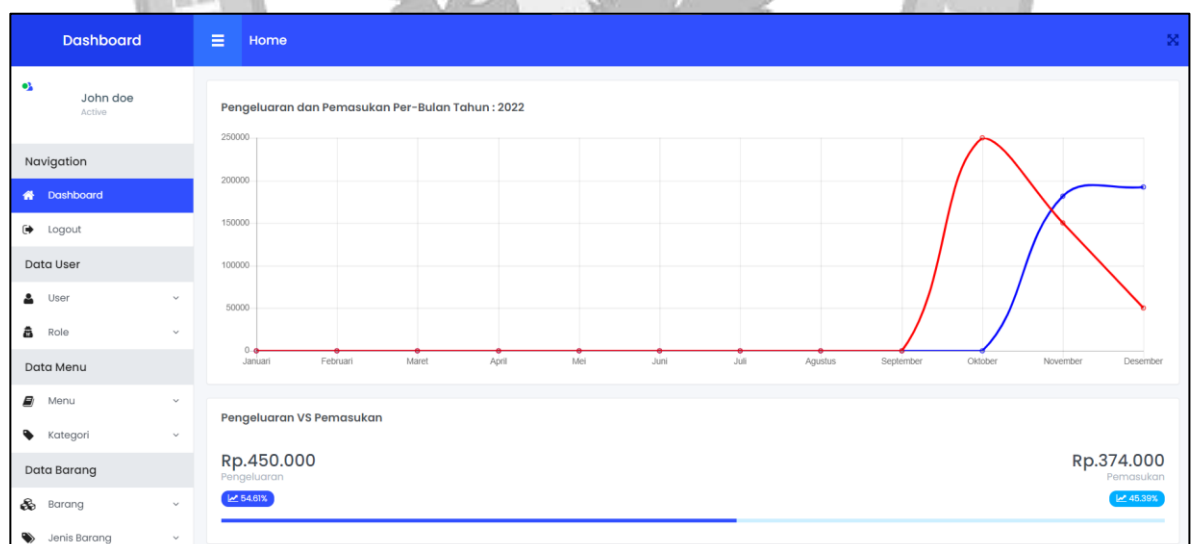
D.2 *Sign Up*

Pada halaman *log in* tersedia link *register account* untuk melakukan registrasi pengguna baru. Pendaftaran ini dapat dilakukan hanya untuk membuat data pengguna kasir atau dapur. Gambar berikut merupakan tampilan dari halaman *sign up*.

Gambar D.2 Halaman *Sign Up*

D.4 *Dashboard Manajer*

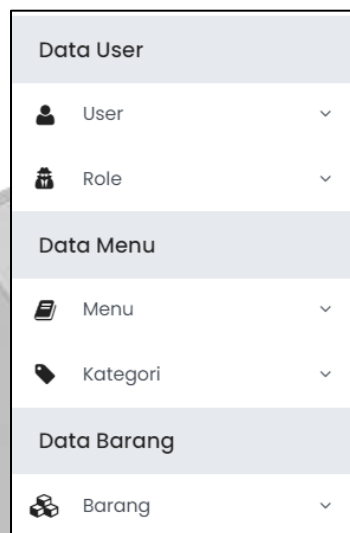
Setelah melakukan *log in*, akan ditampilkan halaman *dashboard* manajer jika pengguna memiliki *role* sebagai manajer. Halaman *dashboard* tersedia beberapa informasi mengenai proses pekerjaan restoran seperti keuangan, inventori, *menu*, dan yang lain-lain. Berikut merupakan tampilan dari halaman *dashboard*.



Gambar D.3 Halaman *Dashboard Manajer*

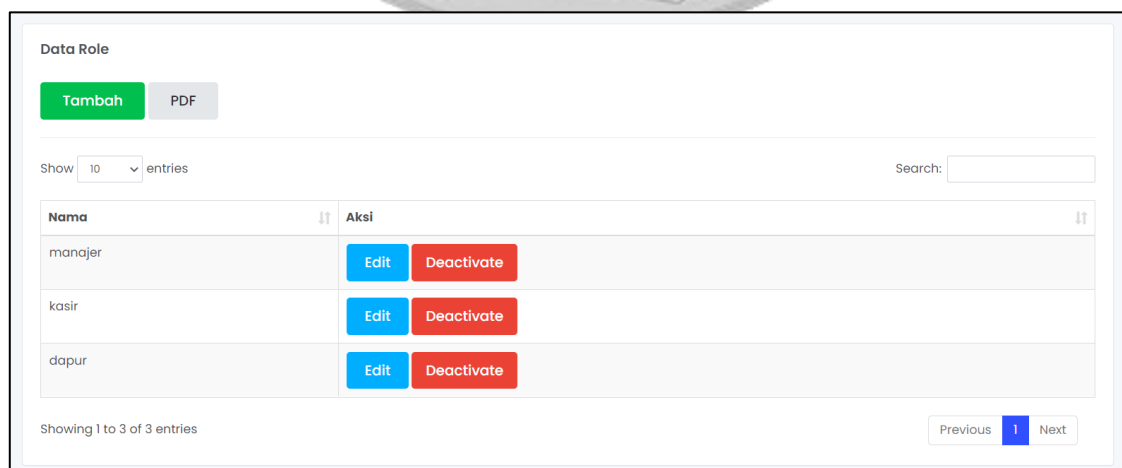
D.5 Master Data

Pada sistem “Restoran Ta Chia” terdapat beberapa halaman *master data* yang hanya bisa diakses oleh manajer. *Master data role*, kategori *menu*, jenis barang, satuan barang, meja, metode bayar, pelanggan, dan *supplier* memiliki fungsi-fungsi yang sama, yaitu menambah, mengubah, dan mendeaktivasi data. Gambar berikut merupakan tampilan dari *sub-menu master data*.



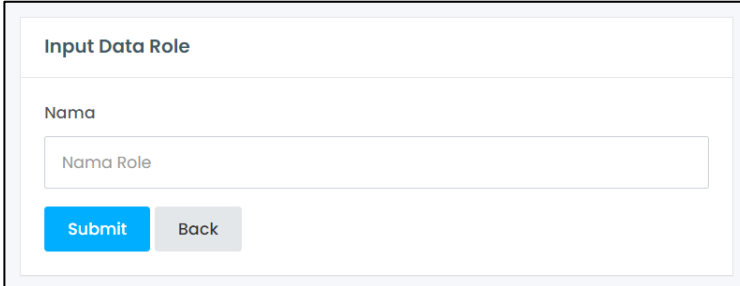
Gambar D.4 Halaman *Side-Nav Master Data*

Gambar berikut merupakan tampilan dari halaman *master data* dari salah satu *sub-menu* yaitu *role*. Pada setiap halaman *master data* telah ditampilkan data-data dalam bentuk tabel dengan beberapa tombol untuk melakukan fungsi-fungsi lainnya. Tombol “Tambah” digunakan untuk melakukan penambahan data baru. Tombol “PDF” digunakan untuk menampilkan halaman PDF. Tombol “*Edit*” digunakan untuk mengubah data. Tombol “*Deactivate*” digunakan untuk mendeaktivasi data. Data yang telah dideaktivasi tidak akan ditampilkan pada tabel.



Gambar D.5 Halaman *Master Data*

Gambar berikut merupakan tampilan dari halaman penambahan data baru. Halaman ubah data memiliki tampilan yang sama tetapi dengan *field* yang sudah terisi dengan data yang sudah ada.



Input Data Role

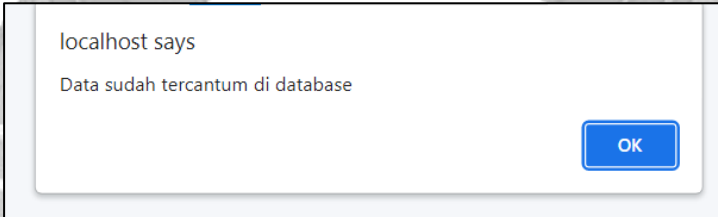
Nama

Nama Role

Submit Back

Gambar D.6 Halaman Tambah Data

Apabila data yang telah diisi dan di-*submit* sudah tersimpan ke dalam *database*, maka akan ditampilkan notifikasi seperti gambar berikut.



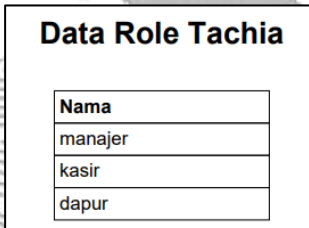
localhost says

Data sudah tercantum di database

OK

Gambar D.7 Notifikasi Duplikasi Data

Berikut merupakan halaman tampilan PDF yang dapat diakses dengan menekan tombol “PDF”.

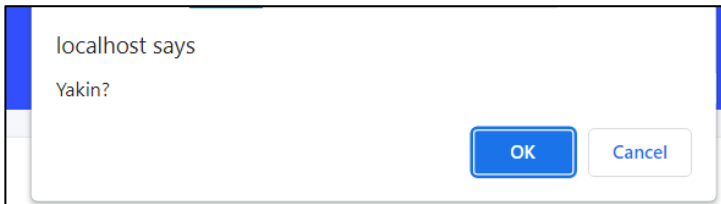


Data Role Tachia

Nama
manajer
kasir
dapur

Gambar D.8 Halaman PDF *Master Data*

Apabila pengguna ingin mendeaktivasi sebuah data, maka akan ditampilkan notifikasi seperti gambar berikut.



localhost says

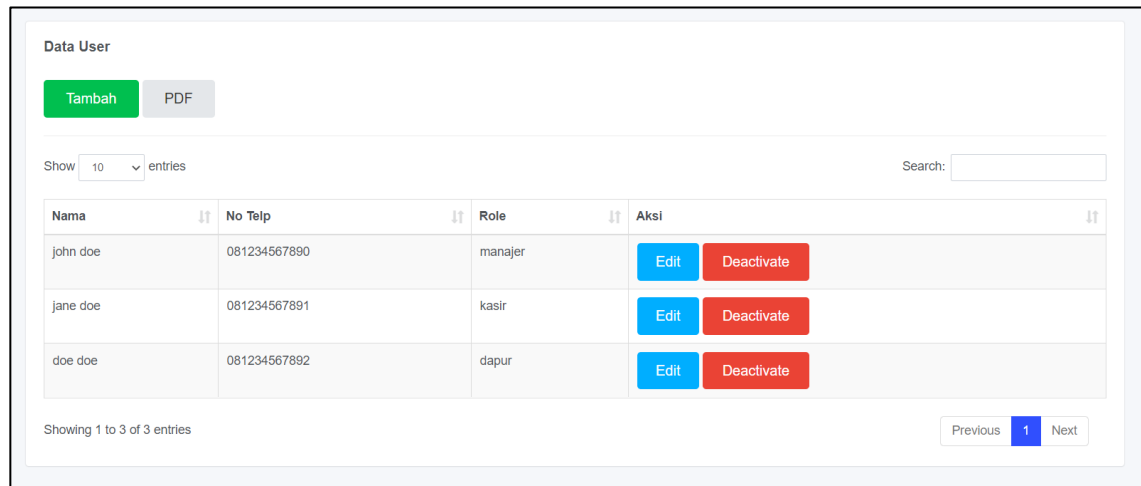
Yakin?

OK Cancel

Gambar D.9 Notifikasi Konfirmasi Deaktivasi Data

D.6 Data User

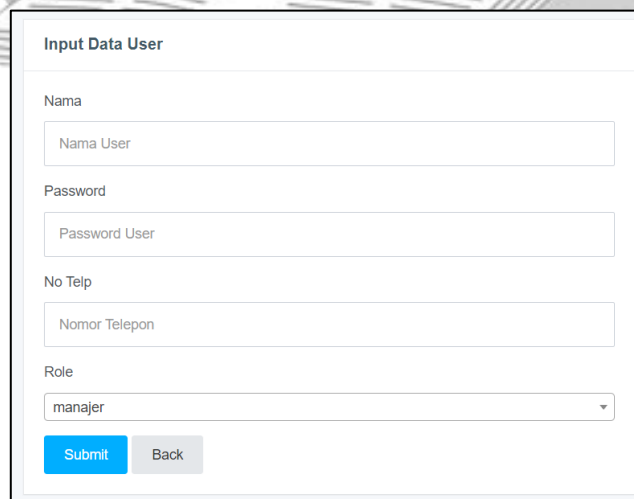
Pada sistem “Restoran Ta Chia”, terdapat fitur untuk mengelola data *user*. Fungsi-fungsi yang tersedia sama dengan fungsi yang dapat dilakukan pada *master data* lainnya, seperti menambah, mengubah, mendeaktivasi, dan menampilkan halaman PDF. Berikut merupakan tampilan dari *master data user*.



Nama	No Telp	Role	Aksi
john doe	081234567890	manajer	Edit Deactivate
jane doe	081234567891	kasir	Edit Deactivate
doe doe	081234567892	dapur	Edit Deactivate

Gambar D.10 Halaman *Master Data User*

Gambar berikut merupakan tampilan dari halaman penambahan data *user*. Sama dengan *master data* yang lain, telah tersedia *field* untuk diisi. Yang membedakan adalah tersedianya *field* pemilihan *role*, yang dimana isinya memerlukan adanya data *role* terlebih dahulu. Halaman *edit user* memiliki tampilan yang sama. Yang membedakannya adalah setiap *field* sudah terisi dengan data yang sesuai. Apabila data yang telah diisi dan di-*submit* sudah tersimpan ke dalam *database*, maka akan ditampilkan notifikasi bahwa data sudah tercantum di *database*.



Input Data User

Nama

Password

No Telp

Role

[Submit](#) [Back](#)

Gambar D.11 Halaman

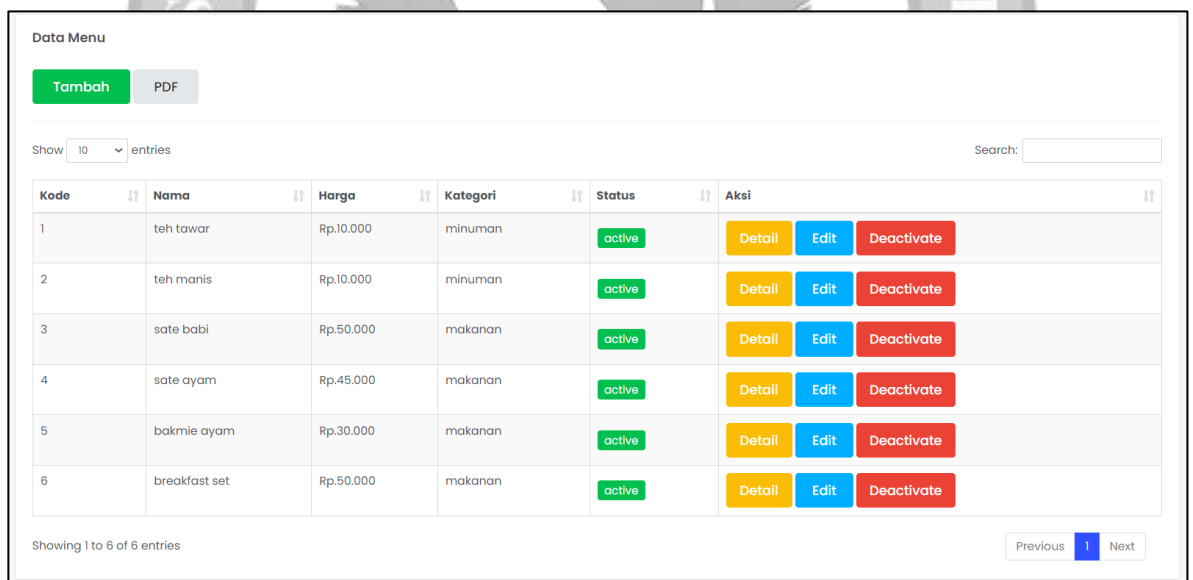
Berikut merupakan halaman tampilan PDF *user* yang dapat diakses dengan menekan tombol “PDF”.

Data User Tachia		
Nama	Nomor Telepon	Role
john doe	081234567890	manajer
jane doe	081234567891	kasir
doe doe	081234567892	dapur

Gambar D.12 Halaman PDF *Master Data User*

D.7 Data Menu

Pada sistem “Restoran Ta Chia”, terdapat fitur untuk mengelola data *menu*. Fungsi-fungsi yang tersedia sama dengan fungsi yang dapat dilakukan pada *master data* lainnya, seperti menambah, mengubah, mendeaktivasi, dan menampilkan halaman PDF. Yang membedakan adalah tersedianya tombol “*Detail*” untuk mengakses *master data menu detail*. Apabila pengguna ingin mendeaktivasi data, status pada tabel akan berubah dari “*active*” menjadi “*inactive*”. Berikut merupakan tampilan dari *master data menu*.



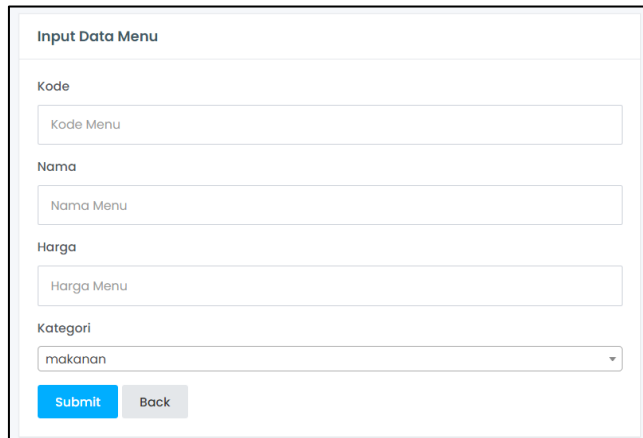
The screenshot shows a web interface titled "Data Menu". At the top left, there are two buttons: "Tambah" (green) and "PDF" (grey). Below these is a "Show 10 entries" dropdown and a search box. The main part of the interface is a table with 6 rows of menu items. Each row has columns for "Kode", "Nama", "Harga", "Kategori", "Status", and "Aksi". The "Status" column shows "active" in a green box. The "Aksi" column contains three buttons: "Detail" (yellow), "Edit" (blue), and "Deactivate" (red). At the bottom left, it says "Showing 1 to 6 of 6 entries". At the bottom right, there are "Previous", "1", and "Next" navigation buttons.

Kode	Nama	Harga	Kategori	Status	Aksi
1	teh tawar	Rp.10.000	minuman	active	Detail Edit Deactivate
2	teh manis	Rp.10.000	minuman	active	Detail Edit Deactivate
3	sate babi	Rp.50.000	makanan	active	Detail Edit Deactivate
4	sate ayam	Rp.45.000	makanan	active	Detail Edit Deactivate
5	bakmie ayam	Rp.30.000	makanan	active	Detail Edit Deactivate
6	breakfast set	Rp.50.000	makanan	active	Detail Edit Deactivate

Gambar D.13 Halaman *Master Data Menu*

Gambar berikut merupakan tampilan dari halaman penambahan data *user*. Sama dengan *master data* yang lain, telah tersedia *field* untuk diisi. Yang membedakan adalah tersedianya *field* pemilihan kategori, yang dimana isinya memerlukan adanya data kategori terlebih dahulu. Halaman *edit menu* memiliki tampilan yang sama. Yang membedakannya adalah setiap *field* sudah terisi dengan data yang sesuai. Apabila data

yang telah diisi dan di-submit sudah tersimpan ke dalam *database*, maka akan ditampilkan notifikasi bahwa data sudah tercantum di *database*.



Input Data Menu

Kode
Kode Menu

Nama
Nama Menu

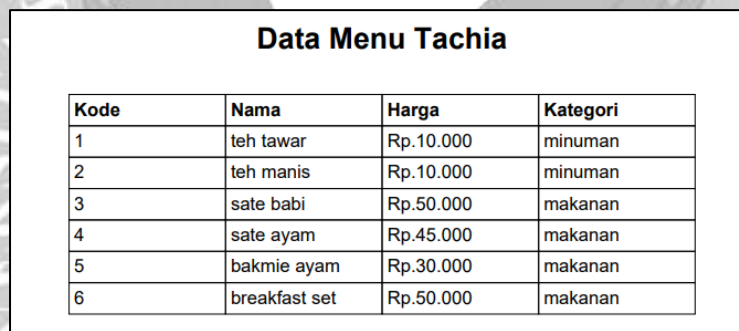
Harga
Harga Menu

Kategori
makanan

Submit Back

Gambar D.14 Halaman Tambah Data Menu

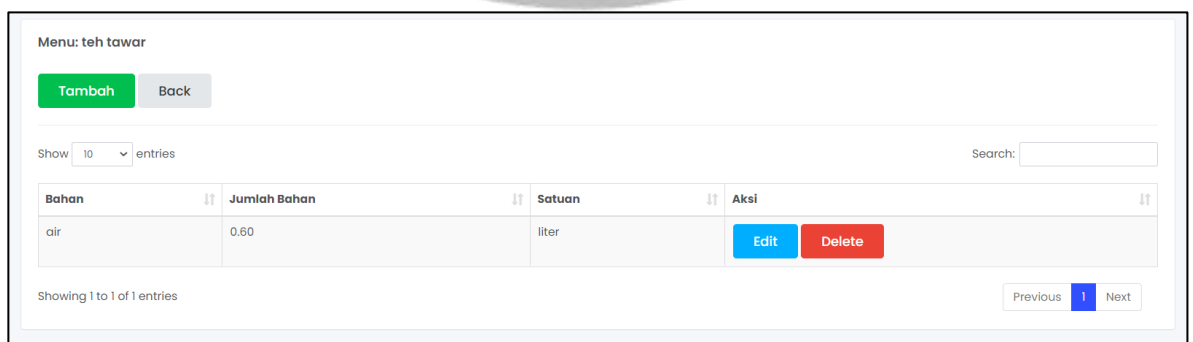
Berikut merupakan halaman tampilan PDF *menu* yang dapat diakses dengan menekan tombol “PDF”.



Kode	Nama	Harga	Kategori
1	teh tawar	Rp.10.000	minuman
2	teh manis	Rp.10.000	minuman
3	sate babi	Rp.50.000	makanan
4	sate ayam	Rp.45.000	makanan
5	bakmie ayam	Rp.30.000	makanan
6	breakfast set	Rp.50.000	makanan

Gambar D.15 Halaman PDF Master Data Menu

Gambar berikut merupakan halaman *menu detail* yang menampilkan bahan-bahan yang diperlukan untuk membuat *menu* tertentu. Fungsi-fungsi yang tersedia sama dengan fungsi pada *master data* lainnya, seperti menambah, mengubah, dan menghapus data.



Menu: teh tawar

Tambah Back

Show 10 entries Search:

Bahan	Jumlah Bahan	Satuan	Aksi
air	0.60	liter	Edit Delete

Showing 1 to 1 of 1 entries Previous 1 Next

Gambar D.16 Halaman Master Data Menu Detail

Gambar berikut merupakan tampilan dari halaman penambahan data *menu detail*. Sama dengan *master data* yang lain, telah tersedia *field* untuk diisi. Yang membedakan adalah tersedianya *field* pemilihan bahan, yang dimana isinya memerlukan adanya data bahan terlebih dahulu. *Field* paling atas menampilkan nama *menu* yang berhubungan. Apabila data yang telah diisi dan di-*submit* sudah tersimpan ke dalam *database*, maka akan ditampilkan notifikasi bahwa data sudah tercantum di *database*.

Gambar D.17 Halaman Tambah Data *Menu Detail*

D.8 Data Barang

Pada sistem “Restoran Ta Chia”, terdapat fitur untuk mengelola data barang. Fungsi-fungsi yang tersedia sama dengan fungsi yang dapat dilakukan pada *master data* lainnya, seperti menambah, mengubah, mendeaktivasi, dan menampilkan halaman PDF. Yang membedakannya adalah tersedianya tombol “Add” yang digunakan untuk menambah jumlah stok barang. Berikut merupakan tampilan dari *master data* barang.

Nama	Jumlah	Jumlah Minimal	Satuan	Jenis Barang	Status	Aksi
sumpit	20.00	5.00	pasang	alat makan	active	Add Edit Deactivate
daging babi	100.00	0.50	kilogram	bahan makan	active	Add Edit Deactivate

Gambar D.18 Halaman *Master Data* Barang

Gambar berikut merupakan tampilan dari halaman penambahan data barang. Sama dengan *master data* yang lain, telah tersedia *field* untuk diisi. Yang membedakan adalah tersedianya *field* pemilihan jenis barang dan satuan, yang dimana isinya

memerlukan adanya data jenis barang dan satuan terlebih dahulu. Halaman *edit* barang memiliki tampilan yang sama. Yang membedakannya adalah setiap *field* sudah terisi dengan data yang sesuai. Apabila data yang telah diisi dan di-*submit* sudah tersimpan ke dalam *database*, maka akan ditampilkan notifikasi bahwa data sudah tercantum di *database*.

Gambar D.19 Halaman Tambah Data Barang

Berikut merupakan halaman tampilan PDF barang yang dapat diakses dengan menekan tombol “PDF”.

Data Barang Tachia			
Nama	Jumlah	Satuan	Kategori
garam	100.00	gram	bahan makan
lada	100.00	gram	bahan makan
tusuk sate	100.00	tangkai	bahan makan
air	100.00	liter	bahan makan
gula	100.00	mililiter	bahan makan
daging ayam	100.00	kilogram	bahan makan
sumpit	20.00	pasang	alat makan
sendok	50.00	butir	alat makan
garpu	50.00	butir	alat makan
daging babi	100.00	kilogram	bahan makan

Gambar D.20 Halaman PDF *Master Data* Barang

Gambar berikut merupakan tampilan dari halaman penambahan jumlah stok barang. Halaman yang ditampilkan memiliki kesamaan seperti halaman *input* dan *edit* data.

Add Jumlah Barang

Nama

Jumlah

Tambah Jumlah

Satuan

Gambar D.21 Halaman Tambah Jumlah Stok Inventori Barang

D.9 Data Pesanan

Pada sistem “Restoran Ta Chia”, terdapat fitur untuk mengelola data pesanan. Beda dengan *master data* lainnya, fungsi-fungsi yang tersedia pada *master data* pesanan hanya ada menampilkan halaman PDF, melakukan *sorting*, dan mengakses *master data* pesanan *detail*. Untuk melakukan *sorting*, pengguna perlu mengisi *field* tanggal awal dan akhir yang tersedia pada sebelah atas tabel dan menekan tombol “Sort”. Isi kedua *field* tanggal sudah terisi dengan tanggal pada hari itu secara *default*. Berikut merupakan tampilan dari *master data* pesanan.

Data Pesanan

Tanggal Awal

Tanggal Akhir

Show entries Search:

Tanggal	Pembayaran	Kembalian	Pemasukan	Urutan	Meja	Pelanggan	Metode Bayar	Diskon	Kasir	Status	Aksi
29/11/2022 - 09:06:45	Rp.60.500	Rp.0	Rp.60.500	1	1	guest	kredit	Tidak Ada	jane doe	sudah bayar	<input type="button" value="Detail"/>
30/11/2022 - 10:46:00	Rp.50.000	Rp.17.000	Rp.33.000	1	1	guest	kas	Tidak Ada	jane doe	sudah bayar	<input type="button" value="Detail"/>
30/11/2022 - 11:17:14	Rp.44.000	Rp.0	Rp.44.000	2	1	steven	debit	Tidak Ada	doe doe	sudah bayar	<input type="button" value="Detail"/>

Gambar D.22 Halaman *Master Data Pesanan*

Berikut merupakan halaman tampilan PDF pesanan yang dapat diakses dengan menekan tombol “PDF”. Pada bagian bawah tabel ditampilkan total dari jumlah pemasukan setiap pesanan yang ditampilkan di dalam tabel.

Data Pesanan Tachia					
Tanggal	Metode Bayar	Diskon	Pembayaran	Kembalian	Pemasukan
29/11/2022 - 09:06:45	kredit	Tidak Ada	Rp.60.500	Rp.0	Rp.60.500
30/11/2022 - 10:46:00	kas	Tidak Ada	Rp.50.000	Rp.17.000	Rp.33.000
30/11/2022 - 03:23:50	kas	Tidak Ada	Rp.20.000	Rp.9.000	Rp.11.000
05/12/2022 - 10:47:47	debit	Tidak Ada	Rp.11.000	Rp.0	Rp.11.000
09/12/2022 - 12:40:59	kredit	Tidak Ada	Rp.33.000	Rp.0	Rp.33.000
10/12/2022 - 09:01:42	kredit	Tidak Ada	Rp.22.000	Rp.0	Rp.22.000
30/11/2022 - 11:17:14	debit	Tidak Ada	Rp.44.000	Rp.0	Rp.44.000
30/11/2022 - 11:17:16	kas	Tidak Ada	Rp.77.000	Rp.44.000	Rp.33.000
12/12/2022 - 12:55:19	debit	Ada	Rp.77.000	Rp.0	Rp.77.000
Total Pemasukan					Rp.324.500

Gambar D.23 Halaman PDF *Master Data Pesanan*

Gambar berikut merupakan halaman pesanan *detail* yang menampilkan *menu-menu* yang telah dipesan pada pesanan tertentu. *Master data pesanan detail* tidak tersedia fungsi-fungsi yang sama dengan yang lain. Pengguna hanya bisa melihat data yang tersedia.

Pesanan #1

Meja 1
13/12/2022 - 08:13:59

[Back](#)

Show 10 entries Search:

Menu	Jumlah	Subtotal	Diskon(%)	Subtotal (Diskon)	Subtotal (Tax)
teh tawar	2	Rp.20.000	0	Rp.20.000	Rp.22.000
sate babi	1	Rp.50.000	50	Rp.25.000	Rp.27.500

Showing 1 to 2 of 2 entries Previous **1** Next

Gambar D.24 Halaman *Master Data Pesanan Detail*

D.10 Data Pengeluaran

Pada sistem “Restoran Ta Chia”, terdapat fitur untuk mengelola data pengeluaran. Fungsi-fungsi yang tersedia sama dengan fungsi yang dapat dilakukan pada *master data* lainnya, seperti menambah, mengubah, *sorting*, dan menampilkan halaman PDF. Untuk melakukan *sorting*, pengguna perlu mengisi *field* tanggal awal dan akhir yang tersedia pada sebelah atas tabel dan menekan tombol “Sort”. Isi kedua *field* tanggal sudah terisi dengan tanggal pada hari itu secara *default*. Berikut merupakan tampilan dari *master data user*.

Data Pengeluaran

Tambah PDF Sort

Tanggal Awal
12/13/2022

Tanggal Akhir
12/13/2022

Show 10 entries Search:

Nama User	Date	Total Pengeluaran	Keterangan	Kategori	Supplier	Aksi
john doe	03/11/2022	Rp.50.000	air	bahan makan	superindo	Edit
john doe	28/11/2022	Rp.100.000	sumpit	alat makan	superindo	Edit
john doe	20/10/2022	Rp.250.000	bumbu bumbu	bahan makan	superindo	Edit
john doe	12/12/2022	Rp.50.000	air	bahan makan	superindo	Edit

Gambar D.25 Halaman *Master Data* Pengeluaran

Gambar berikut merupakan tampilan dari halaman penambahan data pengeluaran. Sama dengan *master* data yang lain, telah tersedia *field* untuk diisi. Yang membedakan adalah tersedianya *field* pemilihan *supplier*, yang dimana isinya memerlukan adanya data *supplier* terlebih dahulu. Halaman *edit* pengeluaran memiliki tampilan yang sama. Yang membedakannya adalah setiap *field* sudah terisi dengan data yang sesuai. Apabila data yang telah diisi dan di-*submit* sudah tersimpan ke dalam *database*, maka akan ditampilkan notifikasi bahwa data sudah tercantum di *database*.

Input Data Pengeluaran

Harga
Total Pengeluaran

Keterangan
Keterangan pengeluaran

Kategori
bahan makan

Supplier
pasar pagi

Tanggal
mm/dd/yyyy

Submit Back

Gambar D.26 Halaman Tambah Pengeluaran

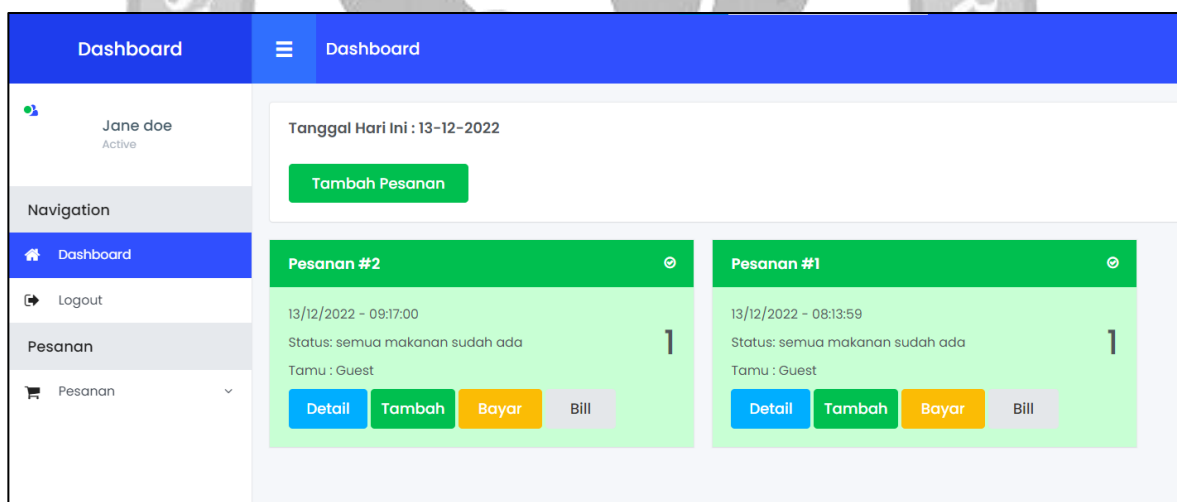
Berikut merupakan halaman tampilan PDF pengeluaran yang dapat diakses dengan menekan tombol “PDF”.

Data Pengeluaran Tachia				
Tanggal	Supplier	Kategori	Keterangan	Total
03/11/2022	superindo	bahan makan	air	Rp.50.000
28/11/2022	superindo	alat makan	sumpit	Rp.100.000
20/10/2022	superindo	bahan makan	bumbu bumbu	Rp.250.000
12/12/2022	superindo	bahan makan	air	Rp.50.000
Total Pengeluaran				Rp.450.000

Gambar D.27 Halaman PDF *Master Data Pengeluaran*

D.11 Menambah Pesanan

Setelah melakukan *log in*, akan ditampilkan halaman *dashboard* kasir jika pengguna memiliki *role* sebagai kasir. Halaman *dashboard* kasir tersedia pesanan-pesanan yang telah dibuat pada hari itu dalam bentuk *card*. Setiap *card* pesanan tersedia beberapa informasi seperti urutan pesanan, waktu pesanan terbuat, status penyajian makanan, nama tamu, dan nomor meja. Pada bagian atas sebelah kanan *card* terdapat *icon* yang menunjukkan status pesanan. Jika pesanan belum selesai, maka *icon* lingkaran kecil akan terlihat kosong. Jika pesanan sudah selesai, maka *icon* lingkaran kecil akan diberikan tanda *checklist*. Pada bagian atas halaman tersedia tombol “Tambah Pesanan” untuk menambah pesanan baru.



Gambar D.28 Halaman *Dashboard Kasir*

Warna yang ditampilkan pada setiap *background card* sudah ditentukan sesuai dengan nomor meja, sedangkan warna yang ditampilkan pada bagian atas *card* ditentukan oleh status penyajian makanan. Jika belum ada makanan disajikan, maka warnanya menjadi abu-abu. Jika beberapa makanan sudah disajikan maka warnanya menjadi kuning. Jika semua makanan sudah disajikan, maka warnanya menjadi hijau.

Terdapat beberapa tombol yang tersedia pada setiap *card*, dan setiap tombol akan tersedia berdasarkan status pesanan. Secara *default*, tombol “Detail” dan

“Tambah” akan selalu tersedia. Jika status berwarna hijau, tombol “Bayar” akan tersedia. Jika sudah melakukan pembayaran, tombol “*Bill*” akan tersedia.

Tombol “Detail” digunakan untuk menampilkan *menu-menu* yang telah dipesan pada pesanan tertentu. Tombol “Tambah” digunakan untuk menambah *menu* pada pesanan. Tombol “Bayar” digunakan untuk melakukan pembayaran. Tombol “*Bill*” digunakan untuk menampilkan *bill* pesanan.



Gambar D.29 Status Pesanan

Berikut merupakan tampilan halaman penambahan pesanan baru. Tersedia *field* pemilihan meja dan pelanggan yang dimana isinya memerlukan adanya data meja dan pelanggan terlebih dahulu.

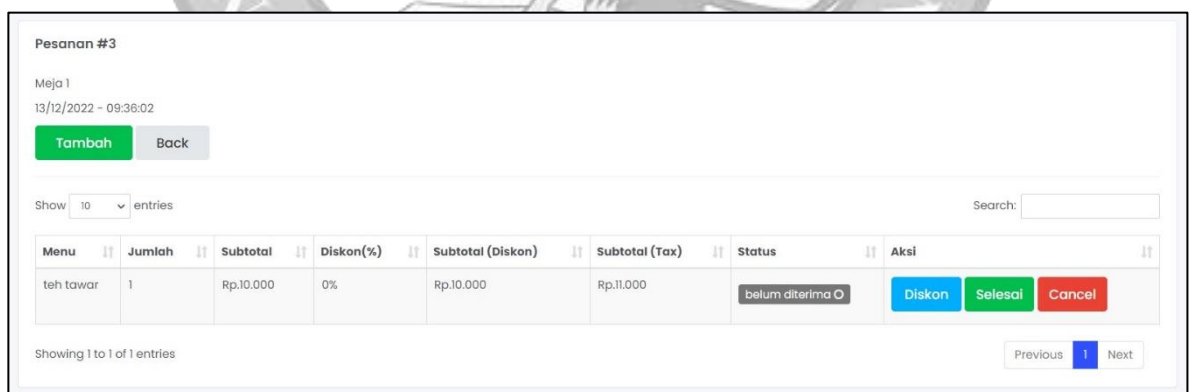


Gambar D.30 Halaman Tambah Pesanan

D.12 Mengelola Pesanan

Berikut merupakan halaman untuk mengelola isi pesanan. Ditampilkan beberapa data mengenai *menu* pada pesanan dalam bentuk tabel. Pada bagian atas halaman tersedia tombol untuk balik ke halaman *dashboard* dan tombol untuk menambah *menu* kepada pesanan. Pada bagian kanan tabel tersedia fungsi-fungsi seperti memberikan diskon, menyelesaikan penyajian *menu*, dan membatalkan pesanan *menu*.

Tombol “Diskon” digunakan untuk memberikan diskon pada *menu* tertentu. Tombol “Selesai” digunakan untuk menyelesaikan status penyajian. Jika status penyajian belum diterima, maka akan ditampilkan notifikasi pesanan belum diterima. Tombol “Cancel” digunakan untuk mengubah status pesanan menjadi “cancelled”. Jika status penyajian sudah diterima, maka akan ditampilkan notifikasi pesanan sudah lagi dimasak.



Menu	Jumlah	Subtotal	Diskon(%)	Subtotal (Diskon)	Subtotal (Tax)	Status	Aksi
teh tawar	1	Rp.10.000	0%	Rp.10.000	Rp.11.000	belum diterima	Diskon Selesai Cancel

Gambar D.31 Halaman Data *Menu* Pada Pesanan

Berikut merupakan tampilan halaman penambahan *menu* pesanan baru yang dapat diakses dengan menekan tombol “Tambah”. Tersedia *field* pemilihan *menu* yang dimana isinya memerlukan adanya data *menu* terlebih dahulu. Jika bahan pada *menu* yang dipilih tidak mencukupi jumlah yang diperlukan, maka akan ditampilkan notifikasi

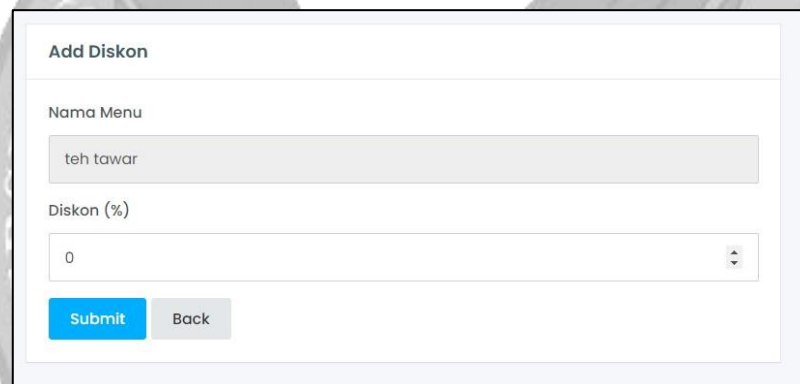
stok barang kurang. Jika pesanan sudah dibayar, maka akan ditampilkan notifikasi pesanan sudah dibayar dan halaman tidak dapat diakses.



The screenshot shows a web form titled "Input Menu Pesanan #3". It contains a "Menu" dropdown menu with "teh tawar" selected, a "Jumlah" input field with the value "1", and two buttons: "Submit" (in blue) and "Back" (in grey).

Gambar D.32 Halaman Tambah *Menu* Pada Pesanan

Berikut merupakan tampilan halaman pemberian diskon yang dapat diakses dengan menekan tombol “Diskon”. Tersedia *field* yang menunjukkan *menu* mana yang akan diberikan diberikan diskon. Jika pesanan sudah dibayar, maka akan ditampilkan notifikasi pesanan sudah dibayar dan halaman tidak dapat diakses.



The screenshot shows a web form titled "Add Diskon". It contains a "Nama Menu" input field with "teh tawar" entered, a "Diskon (%)" input field with "0" entered, and two buttons: "Submit" (in blue) and "Back" (in grey).

Gambar D.33 Halaman Pemberian Diskon

D.13 Konfirmasi Pesanan

Setelah melakukan *log in*, akan ditampilkan halaman *dashboard* dapur jika pengguna memiliki *role* sebagai dapur. Halaman *dashboard* dapur tersedia pesanan *menu* yang telah dibuat pada hari itu dalam bentuk tabel. Pada sebelah kanan tabel tersedia tombol untuk melakukan penerimaan pesanan baru dan penyelesaian masakan.

Tombol “Terima” digunakan untuk menerima pesanan baru, mengubah status pesanan dari “belum diterima” menjadi “lagi dimasak”. Jika status pesanan sudah diterima, maka akan ditampilkan notifikasi pesanan sudah diterima. Tombol “Selesai” digunakan untuk menyelesaikan pesanan, mengubah status pesanan dari “lagi diterima” menjadi “sudah selesai”. Jika status pesanan belum diterima, maka akan ditampilkan notifikasi pesanan belum diterima.

Pesanan	Meja	Menu	Jumlah	Status	Aksi
1	1	safe bati	1	sudah disajikan	Terima Selesai
1	1	teh tawar	2	sudah disajikan	Terima Selesai
2	1	kopi	1	sudah disajikan	Terima Selesai
3	1	teh tawar	1	belum diterima	Terima Selesai

Gambar D.34 Halaman *Dashboard* Dapur

D.14 Pembayaran

Pada halaman *dashboard* kasir terdapat tombol “Bayar” untuk mengakses halaman pada gambar berikut. Halaman berikut menampilkan *bill* penagihan pesanan dan tersedia beberapa fungsi seperti menampilkan halaman PDF, melakukan pembayaran, dan balik ke halaman *dashboard*.

Tombol “*Print*” digunakan untuk menampilkan *bill* penagihan dalam bentuk PDF. Tombol “Bayar” digunakan untuk melakukan pembayaran. Jika pesanan sudah dibayar, maka akan ditampilkan notifikasi pesanan sudah dibayar dan halaman tidak dapat diakses. Tombol “*Home*” digunakan untuk balik ke halaman *dashboard* kasir.

TA-CHIA	
BAKMI SAPI ANGSIO SHI CHUAN	
Jl. Pluit karang timur Blok O-VII Timur No.30-31. Jakarta Utara 14450 Telp. 021-6619086,6601689	
TABLE	3
1 TEH TAWAR	RP.10.000
Net Total	RP.10.000
Pajak 10%	RP.1.000
Total : 1 item	RP.11.000
13/12/2022 - 09:36:02 JANE DOE	
--- BELUM BAYAR ---	
<input type="button" value="Print"/> <input type="button" value="Bayar"/> <input type="button" value="Home"/>	

Gambar D.35 Halaman *Bill* Penagihan

Berikut merupakan tampilan dari halaman pembayaran. Tersedia jumlah harga yang harus dibayar pada bagian atas halaman, dan *field* untuk mengisi jumlah pembayaran dan metode bayar yang dimana isinya memerlukan adanya data metode bayar terlebih dahulu.

Jika metode bayar yang dipilih adalah kas, maka pengguna harus mengisi jumlah pembayaran dengan manual. Jika jumlah pembayaran yang diisi kurang dari *subtotal* pesanan, maka akan ditampilkan notifikasi pembayaran tidak mencukupi. Jika metode bayar yang dipilih adalah yang lain selain kas, maka *field* jumlah pembayaran akan secara otomatis terisi sesuai dengan *subtotal* pesanan.



Subtotal : Rp.11.000,-

Bayar

Pembayaran

Metode Bayar

kas

Submit Back

Gambar D.36 Halaman Pembayaran

Setelah proses pembayaran berhasil, akan ditampilkan halaman *bill* pelunasan. Tersedia tombol “*Print*” untuk menampilkan *bill* pelunasan dalam bentuk PDF, dan tombol “*Home*” untuk balik ke halaman *dashboard* kasir.



TA-CHIA
BAKMI SAPI ANGSI SHI CHUAN

Jl.Pluit karang timur Blok O-VII Timur
No.30-31. Jakarta Utara 14450
Telp. 021-6619086,6601689

TABLE	3
1 TEH TAWAR	RP.10.000
Net Total	RP.10.000
Pajak 10%	RP.1.000
Total : 1 item	RP.11.000
Bayar	RP.20.000
CASH	RP.11.000
Kembali	-RP.9.000

2022-12-13 09:36:02
JANE DOE

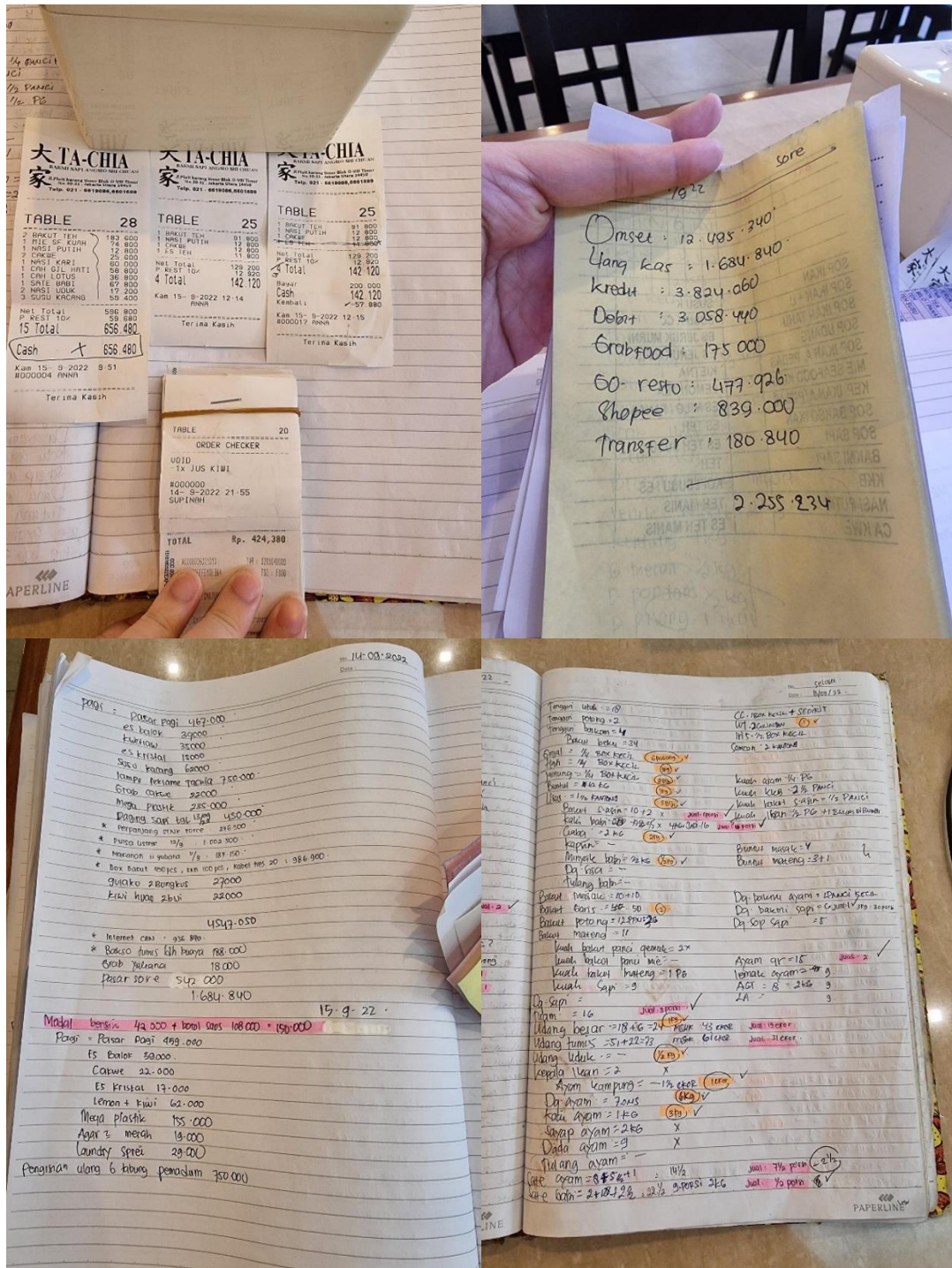
Terima Kasih

Print Home

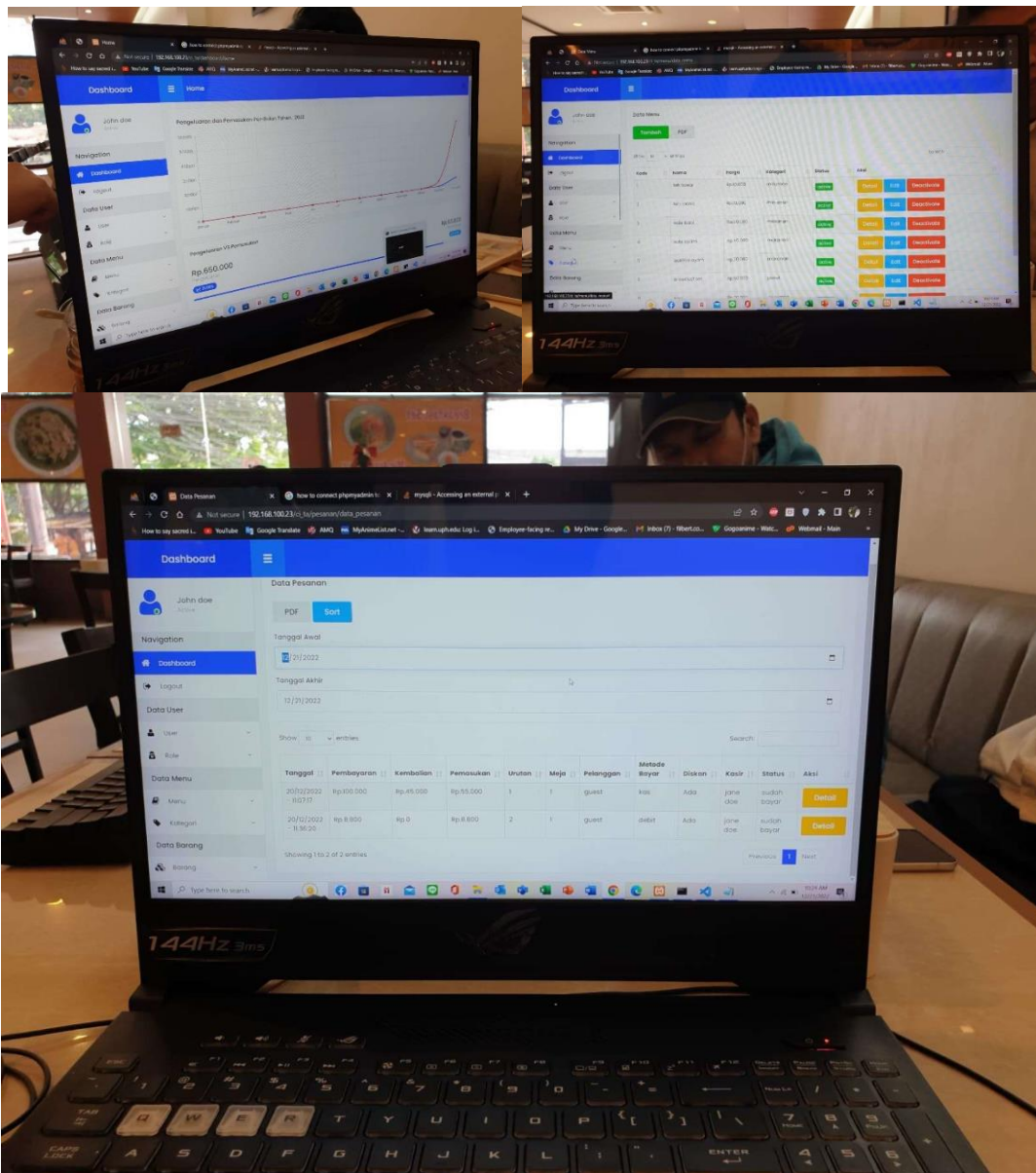
Gambar D.37 Halaman *Bill* Pelunasan

LAMPIRAN E DOKUMENTASI

Dalam mengembangkan sistem pencatatan keuangan dan inventori, pertemuan dan observasi dengan pihak Restoran Ta Chia dilakukan secara langsung pada tempat. Berikut merupakan dokumentasi yang didapatkan selama proses perancangan sistem dibuat.



Gambar E.1 Observasi Pada Restoran Ta Chia



Gambar E.2 Penampilan Sistem Pada Restoran Ta Chia



Gambar E.3 Implementasi Sistem Pada Restoran Ta Chia



UNIVERSITAS PELITA HARAPAN
SISTEM PENJAMINAN MUTU INTERNAL

No. Dok. : FOR03-M/PRO07/STA05/SPMI-UPH

Revisi : 00

Tanggal : 22 Desember 2022

Halaman : 1 dari 1

FORM LEMBAR MONITORING BIMBINGAN TUGAS AKHIR

Nama Mahasiswa / *Student Name* : Filbert Chan
NPM / *Student ID Number* : 01081190011
Program Studi / *Study Program* : Sistem Informasi
Fakultas / *Faculty* : Ilmu Komputer
Lokasi Kampus / *Campus Location* : Karawaci
Semester : 10

Nama Dosen Pembimbing/*Final Project Advisor* : Bapak Kusno

Minggu Ke- <i>Week of-</i>	Hari Day	Tanggal Date	Materi yang didiskusikan <i>The Material Discussed</i>	Tanda Tangan Dosen Pembimbing <i>Final Project Advisor Signature</i>	Catatan <i>Remarks</i>
1	Jumat	8-10-2021	Menentukan jadwal bimbingan		
2	Jumat	22-10-2021	Membahas input dan output dari topik pemilihan Tugas Akhir		
3	Jumat	5-11-2021	Pemutusan penggantian topik Tugas Akhir dan pembahasan input output		
4	Sabtu	13-11-2021	Update progress web; pengecekan password, icon, duplicate, controller		
5	Senin	9-8-2022	Seminar bootcamp		
6	Selasa	20-10-2022	Update progress website, Lanjut ke view pdf dan sistem kasir dapur		
7	Jumat	18-11-2022	Membahas isi dashboard admin, Lanjut benerin bug		
8	Jumat	25-11-2022	Update progress web; Pengurangan bahan baku saat memesan, Lanjut dashboard		
9	Kamis	1-12-2022	Update penyelesaian web, Lanjut pembikinan laporan dari Bab 4-5		
10	Selasa	8-12-2022	Membahas penghubungan phpmyadmin dengan perangkat lain		
11	Minggu	18-12-2022	Revisi laporan Tugas Akhir		
12	Selasa	20-12-2022	Membahas pembuatan video		
12	Rabu	21-12-2022	Set Up Sistem		

Karawaci, 22 Desember 2022

Disetujui oleh/*Approved by:*
Dosen Pembimbing/*Final Project Advisor*

Kusno Prasetya
Tanda Tangan & Nama Lengkap/*Signature & Full Name*

Diketahui oleh/*Acknowledged by:*
Kaprodi/*Department Chair*

Arnold Aribowo
Tanda Tangan & Nama Lengkap/*Signature & Full Name*



UNIVERSITAS PELITA HARAPAN
SISTEM PENJAMINAN MUTU INTERNAL

No. Dok. : FOR03-M/PRO07/STA05/SPMI-UPH

Revisi : 00

FORM LEMBAR MONITORING BIMBINGAN TUGAS AKHIR

Tanggal : 22 Desember 2022

Halaman : 1 dari 1

Nama Mahasiswa / *Student Name* : Filbert Chan
NPM / *Student ID Number* : 01081190011
Program Studi / *Study Program* : Sistem Informasi
Fakultas / *Faculty* : Ilmu Komputer
Lokasi Kampus / *Campus Location* : Karawaci
Semester : 10

Nama Dosen Pembimbing/*Final Project Advisor* : Ibu Calandra

Minggu Ke- <i>Week of-</i>	Hari Day	Tanggal Date	Materi yang didiskusikan <i>The Material Discussed</i>	Tanda Tangan Dosen Pembimbing <i>Final Project Advisor Signature</i>	Catatan <i>Remarks</i>
1	Jumat	22-10-2021	Membahas input dan output dari topik pemilihan Tugas Akhir		
2	Jumat	5-11-2021	Pemutusan penggantian topik Tugas Akhir dan pembahasan input output		
3	Senin	8-8-2022	Seminar bootcamp		
4	Rabu	14-9-2022	Pembenaran Bab 1-3, Pembahasan sistem database proyek		
5	Kamis	22-9-2022	Pembahasan sistem database proyek (lanjut)		
6	Kamis	29-9-2022	Ganti template baru, Update progress pembenaran Bab 1-3		
7	Kamis	6-10-2022	Update progress penginputan data pada web, Lanjut ke Bab 4		
8	Kamis	13-10-2022	Update progress web; pengecekan password, icon, duplicate, controller		
9	Kamis	20-10-2022	Update progress website, Lanjut ke view pdf dan sistem kasir dapur		
10	Kamis	3-11-2022	Membahas flow pemesanan kasir dapur		
11	Sabtu	12-11-2022	Membahas masalah session login (Makrab)		
12	Jumat	18-11-2022	Membahas isi dashboard admin, Lanjut benerin bug		
13	Kamis	1-12-2022	Update penyelesaian web, Lanjut pembikinan laporan dari Bab 4-5		

Karawaci, 22 Desember 2022

Disetujui oleh/*Approved by:*
Dosen Pembimbing/*Final Project Advisor*

Calandra Alencia Haryani
Tanda Tangan & Nama Lengkap/*Signature & Full Name*

Diketahui oleh/*Acknowledged by:*
Kaprosdi/*Department Chair*

Arnold Aribowo
Tanda Tangan & Nama Lengkap/*Signature & Full Name*