

**IMPLEMENTASI *FRONT-END* PADA DIGITALISASI PEMESANAN  
MENU MAKANAN DAN MINUMAN BERBASIS *WEBSITE*  
(STUDI KASUS: TOKO MANSURE)**

**(Skripsi)**

**Oleh**

**OKTA SILVI  
NPM 1957051009**



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS LAMPUNG  
BANDAR LAMPUNG  
2023**

## ABSTRAK

### IMPLEMENTASI *FRONT-END* PADA DIGITALISASI PEMESANAN MENU MAKANAN DAN MINUMAN BERBASIS *WEBSITE* (STUDI KASUS: TOKO MANSURE)

Oleh

OKTA SILVI

Toko Mansure merupakan salah satu kafe yang terletak di Bandar Lampung, yang bergerak dalam bidang FnB. Saat ini, Toko Mansure masih menerapkan sistem manual pada proses pemesanan makanan dan minumannya. Proses pemesanan dilakukan langsung di bagian kasir saat pelanggan datang. Informasi pada daftar menu yang disediakan masih belum lengkap, seperti deskripsi menu dan masih menggunakan kertas. Pada saat pembayaran, kasir mencetak 2 setruk, yang dimana satu untuk pelanggan dan satu untuk bagian koki untuk memproses pesanan pelanggan. Estimasi total pembayaran yang harus dibayar pelanggan belum bisa dilihat langsung oleh pelanggan. Akibatnya, terdapat beberapa permasalahan pada sistem yang berjalan saat ini, yaitu banyaknya pengeluaran yang harus dikeluarkan dalam penggunaan kertas setruk dan kertas daftar menu jika ada pembaharuan menu dan juga pelanggan harus menghitung estimasi total pembayaran dahulu karena belum bisa melihat estimasi total pembayaran secara langsung.

Pemanfaatan teknologi informasi berbasis *website* dianggap lebih baik karena pelanggan dapat melakukan pemesanan melalui *gadget* masing-masing. Oleh karena itu, penelitian ini bertujuan untuk membangun *front-end* pada digitalisasi pemesanan makanan dan minuman berbasis *website*. Sistem berbasis *website* ini dikembangkan menggunakan metode *extreme programming* dengan bahasa pemrograman PHP, HTML, CSS dan JavaScript serta *framework CodeIgniter 4* dan *bootstrap 5*. Sistem ini sudah dilakukan pengujian dengan pengujian *responsiveness* dengan *tools ResponsivelyApp*, pengujian kecepatan dengan *tools lighthouse*, pengujian kemudahan pengguna menggunakan metode *usability testing* dengan kuesioner SUS dan pengujian aksesibilitas menggunakan *aXe DevTools*.

Kata kunci: Sistem informasi, *extreme programming*, digitalisasi, *front-end*.

## **ABSTRACT**

### **BACK-END IMPLEMENTATION ON WEBSITE-BASED FOOD AND BEVERAGE MENU ORDERING DIGITALIZATION (CASE STUDY: TOKO MANSURE)**

**By**

**OKTA SILVI**

Toko Mansure is one of the cafes located in Bandar Lampung, which is engaged in FnB. Currently, Toko Mansure still applies a manual system to the food and beverage ordering process. The ordering process is done directly at the cashier when the customer arrives. The information on the menu list provided is still incomplete, such as menu descriptions and still uses paper. At the time of payment, the cashier prints 2 receipts, one for the customer and one for the chef to process the customer's order. The estimated total payment that must be paid by the customer cannot be seen directly by the customer. As a result, there are several problems with the current system, namely the amount of expenses that must be incurred in using paper receipts and paper menu lists if there is a menu update and also customers must calculate the total estimated payment first because they cannot see the total estimated payment directly.

The utilization of website-based information technology is considered better because customers can place orders through their respective gadgets. Therefore, this research aims to build a front-end on the digitization of web-based food and beverage ordering. This web-based system was developed using the extreme programming method with PHP, HTML, CSS and JavaScript programming languages as well as the CodeIgniter 4 framework and bootstrap 5. This system has been tested with responsiveness testing with ResponsivelyApp tools, speed testing with lighthouse tools, user-friendliness testing using usability testing methods with SUS questionnaires and accessibility testing using aXe DevTools.

*Keywords: information system, extreme programming, digitalization, front-end.*

**IMPLEMENTASI *FRONT-END* PADA DIGITALISASI PEMESANAN  
MENU MAKANAN DAN MINUMAN BERBASIS *WEBSITE*  
(STUDI KASUS: TOKO MANSURE)**

**Oleh**

**OKTA SILVI**

**Skripsi**

**Sebagai Salah Satu Syarat untuk Mencapai Gelar  
SARJANA KOMPUTER**

**Pada**

**Jurusan Ilmu Komputer  
Fakultas Matematika dan Ilmu Pengetahuan Alam  
Universitas Lampung**



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS LAMPUNG  
BANDAR LAMPUNG  
2023**

Judul Skripsi : **IMPLEMENTASI FRONT-END PADA DIGITALISASI PEMESANAN MENU MAKANAN DAN MINUMAN BERBASIS WEBSITE (STUDI KASUS: TOKO MANSURE)**

Nama Mahasiswa : Okta Silvi


Nomor Pokok Mahasiswa : 1957051009

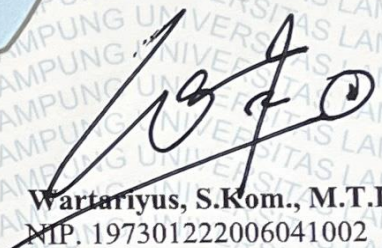
Program Studi : S1 Ilmu Komputer

Fakultas : Matematika dan Ilmu Pengetahuan Alam



1. Komisi Pembimbing

  
Yohana Tri Utami, S.Kom., M.Kom.  
NIP. 199001102019032010

  
Wartariyus, S.Kom., M.T.I.  
NIP. 197301222006041002

2. Ketua Jurusan Ilmu Komputer

  
Didik Kurniawan, S.Si., M.T.  
NIP. 198004192005011004

**MENGESAHKAN**

**1. Tim Penguji**

**Ketua : Yohana Tri Utami, S.Kom., M.Kom.**

**Penguji I  
Sekretaris : Wartariyus, S.Kom., M.T.I.**

**Penguji II  
Bukan Pembimbing : Didik Kurniawan, S.Si, M.T.**

**2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam**



**Dr. Ing. Heri Satria, S.Si., M.Si.**  
**NIP. 197110012005011002**

**Tanggal Lulus Ujian Skripsi : 26 Juli 2023**

## PERNYATAAN

Saya yang bertanda tangan di bawah ini, menyatakan bahwa skripsi saya yang berjudul **“Implementasi *Front-End* Pada Digitalisasi Pemesanan Menu Makanan Dan Minuman Berbasis *Website* (Studi Kasus: Toko Mansure)”** merupakan hasil karya sendiri dan bukan hasil karya orang lain. Semua hasil yang tertuang dalam skripsi ini telah mengikuti kaidah penulisan karya ilmiah Universitas Lampung. Apabila di kemudian hari terbukti skripsi ini merupakan salinan atau dibuat oleh orang lain, maka saya bersedia menerima sanksi sesuai dengan ketentuan akademik yang berlaku.

Bandar Lampung, 26 Juli 2023



Okta Silvi  
NPM 1957051009

## RIWAYAT HIDUP



Penulis dilahirkan di Bandar Lampung pada hari Senin, 22 Oktober 2001, sebagai anak kelima dari lima bersaudara, dari Bapak Sunarto dan Ibu Noparti. Pendidikan Sekolah Dasar (SD) diselesaikan pada tahun 2013 di SDN 1 Rawa Laut. Kemudian Pendidikan menengah pertama diselesaikan pada tahun 2016 di SMP Negeri 5 Bandar Lampung dan pendidikan menengah atas diselesaikan di SMA YP UNILA pada tahun 2019.

Pada tahun 2019, penulis terdaftar sebagai mahasiswa Jurusan Ilmu Komputer Universitas Lampung melalui jalur SMMPTN-Barat. Selama menjadi mahasiswa, penulis pernah melakukan kegiatan-kegiatan, yaitu sebagai berikut.

1. Mengikuti Karya Wisata Ilmiah (KWI) FMIPA pada tahun 2019 di Desa Tambah Dadi, Kecamatan Purbolinggo, Lampung Timur, Lampung.
2. Menjadi Anggota Bidang Keilmuan Himpunan Mahasiswa Jurusan Ilmu Komputer periode 2020/2021.
3. Menjadi Bendahara Bidang Keilmuan Himpunan Mahasiswa Jurusan Ilmu Komputer periode 2021/2022.
4. Menjadi Asisten Dosen Jurusan Ilmu Komputer pada tahun 2021.



5. Mengikuti Kuliah Kerja Nyata 2022 periode I di Kelurahan Sawah Lama, Bandar Lampung.
6. Mengikuti Praktek Kerja Lapangan (PKL) 2022 periode II di Kementerian Agama Prov. Lampung.
7. Mengikuti magang mandiri di *Pro-Strategic* sebagai *Search Engine Optimization* (SEO) pada tahun 2023.

## **MOTTO**

“Maka bersabarlah kamu, sesungguhnya janji Allah itu benar.”

(QS. Al-Ghafir:55)

“Sesukses apapun kamu, selalu ingatlah kedua orang tuamu:).”

(Anonim)

## **PERSEMBAHAN**

Puji syukur Kehadirat Allah SWT, karena atas rahmat dan karunia-Nya sehingga dapat menyelesaikan skripsi ini dengan sebaik-baiknya.

Kupersembahkan karya sederhana ini kepada:

Ayah dan Ibuku tercinta

*Ayahanda Sunarto dan Ibunda Noparti*

Saudaraku tersayang

*Ari Pratama, Ardian, Arphin, Ivan Lee, Devi Ika dan Nendy Utari*

Yang selalu mendoakan serta mendukung setiap langkah yang saya lalui. Saya ucapkan terimakasih sebesar-besarnya atas kasih sayang, didikan serta pengorbanan yang tak terhingga yang telah diberikan kepada saya.

**Seluruh Keluarga Besar Ilmu Komputer 2019**

**Jurusan Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam,**

**Universitas Lampung**

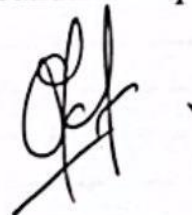
## SANWACANA

Puji Syukur Kehadirat Allah SWT atas rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan skripsi yang berjudul “**Implementasi *Front-End* Pada Digitalisasi Pemesanan Menu Makanan Dan Minuman Berbasis *Website* (Studi Kasus: Toko Mansure)**” dengan sebaik-baiknya. Skripsi ini adalah salah satu syarat untuk memperoleh gelar sarjana Ilmu Komputer di Universitas Lampung. Dalam kesempatan ini, penulis banyak mendapat bimbingan, dukungan serta doa dari berbagai pihak, sehingga penulis ingin menyampaikan ungkapan terima kasih ini kepada:

1. Keluarga, Ayahanda Sunarto dan Ibunda Noparti yang selalu memberikan doa serta dukungan pada setiap langkah yang saya lalui.
2. Bapak Dr. Eng. Heri Satria, S.Si., M.Si. selaku Dekan FMIPA Universitas Lampung.
3. Bapak Didik Kurniawan, S.Si., M.T. selaku Ketua Jurusan Ilmu Komputer Universitas Lampung sekaligus Dosen Pembahas yang telah memberikan masukan dalam penelitian skripsi ini.
4. Dr. rer. Nat. Akmal Junaidi, M.Sc. selaku Sekretaris Jurusan Ilmu Komputer Universitas Lampung.
5. Bapak Rangga Firdaus, M.Kom. selaku Dosen Pembimbing Akademik.
6. Ibu Yohana Tri Utami, S.Kom., M.Kom. selaku Dosen Pembimbing utama yang telah membimbing saya sehingga dapat menyelesaikan skripsi ini dengan baik.
7. Bapak Wartariyus, S.Kom., M.T.I. selaku Dosen Pembimbing kedua yang telah memberi masukan dalam proses menyelesaikan skripsi.

8. Para Bapak dan Ibu Dosen Jurusan Ilmu Komputer FMIPA Universitas Lampung yang telah memberikan ilmu serta pengalaman semasa saya berkuliah.
9. Ibu Ade Noera Maela, Mas Noval, Bang Zainuddin yang telah membantu dalam hal administrasi di Jurusan Ilmu Komputer.
10. Pihak Toko Mansure yang memberikan kesempatan untuk melakukan penelitian skripsi ini.
11. Nadia Salsabila selaku rekan skripsi yang selalu mendukung serta memberi semangat dalam proses menyelesaikan skripsi.
12. Grup Himatuy (Nabilah Putri, Qurrota Aini, Nadia Salsabila, Ghina Aramita, Lulu Agustin) serta teman-teman S1 Ilmu Komputer Angkatan 2019 yang telah memberikan pengalaman semasa saya berkuliah.
13. Indri Mustika, Winda Ayu, Nabila Ferdiana, Grup Assalamualaikum (Hanifah Viralina, Farani Tasya, Made Chendy, Rizky Ayu, Niwayan Lisa Putri) yang telah menemani *healing*, membantu serta mendukung dalam proses menyelesaikan skripsi.

Bandar Lampung, 26 Juli 2023



Okta Silvi

NPM. 1957051009

## DAFTAR ISI

Halaman

<b>ABSTRAK .....</b>	<b>iv</b>
<b>ABSTRACT .....</b>	<b>v</b>
<b>DAFTAR ISI.....</b>	<b>vi</b>
<b>DAFTAR TABEL.....</b>	<b>viii</b>
<b>DAFTAR GAMBAR.....</b>	<b>ix</b>
<b>I. PENDAHULUAN.....</b>	<b>1</b>
1.1. Latar Belakang .....	1
1.2. Rumusan Masalah.....	3
1.3. Batasan Masalah .....	3
1.4. Tujuan Penelitian .....	4
1.5. Manfaat Penelitian .....	4
<b>II. TINJAUAN PUSTAKA .....</b>	<b>5</b>
2.1. Penelitian Terdahulu .....	5
2.2. Uraian Landasan Teori.....	8
2.2.1. Pengertian Sistem .....	8
2.2.2. Pengertian Informasi.....	8
2.2.3. Pengertian Sistem Informasi .....	9
2.2.4. Pengertian Digitalisasi .....	9
2.2.5. Pengertian <i>Website</i> .....	9
2.2.6. Pengertian HTML .....	10
2.2.7. Pengertian CSS .....	11
2.2.8. Pengertian CodeIgniter .....	11
2.2.9. Pengertian <i>Unified Modeling Language (UML)</i> .....	12
2.2.10. Pengertian XP ( <i>Extreme Programming</i> ).....	14
2.2.11. Pengertian <i>Front-End</i> .....	15
2.2.12. Pengertian <i>Responsively App</i> .....	17
2.2.13. Pengertian <i>Usability</i> (Kegunaan).....	17
2.2.14. Pengertian <i>Usability Testing</i> .....	17
2.2.15. Pengertian <i>System Usability Scale</i> .....	18
2.2.16. Pengertian <i>Lighthouse</i> .....	19
2.2.17. Pengertian AXE DevTools .....	20
<b>III. METODE PENELITIAN .....</b>	<b>21</b>
3.1. Waktu dan Tempat Penelitian.....	21

3.2.	Tahapan Penelitian.....	21
3.2.1.	Tahapan Pengumpulan Data.....	22
3.2.2.	Tahapan Pengembangan Sistem .....	23
3.2.2.1.	Perencanaan.....	23
3.2.2.2.	Desain.....	25
3.2.2.3.	Pengkodean.....	44
3.2.2.4.	Pengujian.....	44
3.2.3.	Tahapan Penulisan Laporan.....	51
<b>IV.</b>	<b>Hasil dan Pembahasan .....</b>	<b>53</b>
4.1.	Hasil.....	53
4.2.	Pembahasan .....	54
4.2.1.	Tahapan Pengkodean .....	54
4.2.2.	Tahapan Pengujian.....	71
<b>V.</b>	<b>KESIMPULAN DAN SARAN .....</b>	<b>91</b>
5.1.	Kesimpulan.....	91
5.2.	Saran .....	91
	<b>DAFTAR PUSTAKA .....</b>	<b>93</b>

## DAFTAR TABEL

	Halaman
Tabel 1. Simbol-Simbol <i>Use Case Diagram</i> .....	13
Tabel 2. Simbol-Simbol <i>Activity Diagram</i> .....	13
Tabel 3. Skenario Pengujian Modul Pelanggan .....	45
Tabel 4. Skenario Pengujian Modul Koki.....	46
Tabel 5. Skenario Pengujian Modul Kasir .....	46
Tabel 6. Skenario Pengujian Modul Super Admin .....	47
Tabel 7. Instrumen SUS .....	48
Tabel 8. Kriteria Penilaian .....	49
Tabel 9. Evaluasi Parameter <i>Performance Metric</i> .....	50
Tabel 10. Alur Waktu Tahapan Penelitian.....	52
Tabel 11. Variasi Responden .....	77
Tabel 12. Rekap Jawaban Responden.....	78
Tabel 13. Rekapitulasi Jawaban Responden Setelah Perhitungan SUS.....	78
Tabel 14. Nilai Akhir SUS .....	79
Tabel 15. Hasil Pengukuran <i>Performance Metric</i> .....	81
Tabel 16. Hasil Pengujian Aksesibilitas Tampilan Pelanggan.....	82
Tabel 17. Hasil Pengujian Aksesibilitas Tampilan Admin. ....	83
Tabel 18. Kategori Dampak Terhadap Pengguna. ....	84
Tabel 19. Hasil Pengujian Aksesibilitas Tampilan Pelanggan Terbaru. ....	90
Tabel 20. Hasil Pengujian Aksesibilitas Tampilan Admin Terbaru. ....	90



## DAFTAR GAMBAR

	Halaman
Gambar 1. Metodologi XP.....	14
Gambar 2. Penentuan Hasil Evaluasi <i>Usability</i> dengan SUS.....	19
Gambar 3. Tahapan Penelitian.....	22
Gambar 4. <i>Use Case Diagram</i> .....	25
Gambar 5. <i>Activity Diagram</i> Akses Beranda.....	26
Gambar 6. <i>Activity Diagram</i> Pesan Menu.....	26
Gambar 7. <i>Activity Diagram</i> Tagihan Pembayaran.....	27
Gambar 8. <i>Activity Diagram Login</i> Super Admin.....	27
Gambar 9. <i>Activity Diagram Login</i> Kasir.....	28
Gambar 10. <i>Activity Diagram Login</i> Koki.....	28
Gambar 11. <i>Activity Diagram</i> Tambah Menu.....	29
Gambar 12. <i>Activity Diagram</i> Edit Menu.....	30
Gambar 13. <i>Activity Diagram</i> Hapus Menu.....	30
Gambar 14. <i>Activity Diagram</i> Melihat dan Mengubah Status Pesanan.....	31
Gambar 15. <i>Activity Diagram</i> Melihat dan Mengubah Status Pesanan Bagian Kasir.....	31
Gambar 16. <i>Activity Diagram</i> Mencetak Laporan Pembayaran.....	32
Gambar 17. <i>Activity Diagram</i> Tambah <i>User</i> .....	33
Gambar 18. <i>Activity Diagram</i> Hapus <i>User</i> .....	33
Gambar 19. Rancangan Tampilan Beranda.....	34
Gambar 20. Rancangan Tampilan Menu.....	35
Gambar 21. Rancangan Tampilan Keranjang Pesanan.....	35
Gambar 22. Rancangan Tampilan Tagihan Pembayaran.....	36
Gambar 23. Rancangan Tampilan <i>Login</i> Karyawan.....	37

Gambar 24. Rancangan Tampilan <i>Dashboard</i> Super Admin. ....	37
Gambar 25. Rancangan Tampilan Daftar Menu. ....	38
Gambar 26. Rancangan Tampilan Hapus Menu. ....	38
Gambar 27. Rancangan Tampilan Tambah Menu. ....	39
Gambar 28. Rancangan Tampilan Edit Menu. ....	39
Gambar 29. Rancangan Tampilan <i>User Role</i> . ....	40
Gambar 30. Rancangan Tampilan Hapus Akun <i>User</i> . ....	40
Gambar 31. Rancangan Tampilan Reset <i>Password User</i> . ....	41
Gambar 32. Rancangan Tampilan Koki. ....	41
Gambar 33. Rancangan Tampilan Kasir. ....	42
Gambar 34. Rancangan Tampilan Transaksi Pembayaran. ....	42
Gambar 35. Rancangan Tampilan Setruk Pembelian. ....	43
Gambar 36. Rancangan Tampilan Cetak Laporan. ....	43
Gambar 37. Rancangan Tampilan Hasil Export Laporan. ....	44
Gambar 38. <i>Parameter Score</i> . ....	50
Gambar 39. Halaman Beranda Pelanggan. ....	54
Gambar 40. Halaman Menu. ....	55
Gambar 41. Halaman Keranjang Pesanan. ....	56
Gambar 42. Halaman Tagihan Pembayaran. ....	57
Gambar 43. Halaman <i>Login</i> Karyawan. ....	58
Gambar 44. Halaman <i>Dashboard</i> Super Admin. ....	59
Gambar 45. Halaman Daftar Menu. ....	60
Gambar 46. Halaman Hapus Menu. ....	61
Gambar 47. Halaman Tambah Menu. ....	62
Gambar 48. Halaman Edit Menu. ....	63
Gambar 49. Halaman <i>User Role</i> . ....	64
Gambar 50. Halaman Hapus Akun <i>User</i> . ....	64
Gambar 51. Halaman Reset <i>Password User</i> . ....	65
Gambar 52. Halaman Koki. ....	66
Gambar 53. Halaman Kasir. ....	67
Gambar 54. Halaman Transaksi Pembayaran. ....	68
Gambar 55. Halaman Setruk Pembelian. ....	69

Gambar 56. Halaman Laporan. ....	70
Gambar 57. Halaman <i>Export PDF</i> . ....	71
Gambar 58. <i>Grid Layout</i> Menu pada Perangkat Laptop Macbook Pro. ....	72
Gambar 59. <i>Grid Layout</i> Menu pada Perangkat Ponsel iPhone 14.....	72
Gambar 60. <i>Grid Layout</i> Menu pada Perangkat Ponsel Galaxy S20 Plus. ....	73
Gambar 61. <i>Grid Layout</i> Menu pada Perangkat Tablet iPad Mini. ....	73
Gambar 62. <i>Media queries</i> pada Perangkat Laptop Macbook Pro. ....	74
Gambar 63. <i>Media queries</i> pada Perangkat Ponsel iPhone 14.....	75
Gambar 64. <i>Media queries</i> pada Perangkat Ponsel Galaxy S20 Plus.....	75
Gambar 65. <i>Media queries</i> pada Perangkat Tablet iPad Mini. ....	76
Gambar 66. Kegiatan Pengujian Sistem.....	77
Gambar 67. Hasil Pengujian Kecepatan.....	80
Gambar 68. Perubahan Tampilan pada Halaman Pelanggan.....	85
Gambar 69. Perubahan Tampilan pada Navbar Pelanggan.....	86
Gambar 70. Perubahan Tampilan pada Halaman Menu. ....	87
Gambar 71. Perubahan Tampilan Halaman Keranjang Saya.....	88
Gambar 72. Perubahan Tampilan pada Halaman Kasir. ....	89

## I. PENDAHULUAN

### 1.1. Latar Belakang

Perkembangan teknologi informasi pada zaman sekarang sudah semakin canggih, tak terkecuali pada dunia bisnis. Perkembangan teknologi informasi banyak dibutuhkan oleh seluruh pengusaha untuk mengelola perusahaannya agar dapat bersaing di dunia perbisnisan. Saat ini, sudah banyak kafe yang sering kita temui, sehingga terjadinya persaingan antar kafe. Untuk meningkatkan daya saing di dunia perbisnisan maka perusahaan membutuhkan sistem pelayanan yang baik terhadap pelanggan-nya sehingga mampu meningkatkan kualitas yang besar terhadap proses bisnis yang terjadi.

Toko Mansure merupakan sebuah kafe yang menjual makanan dan minuman serta menyuguhkan tempat yang bernuansa santai. Toko Mansure beralamat di Jalan KH. Mas Mansyur, Rawa Laut, Bandar Lampung. Toko Mansure memiliki konsep bangunan ala *Scandinavian house* yang dimana memiliki 3 *spot* yang ditawarkan, yaitu *outdoor*, *semi indoor* dan *full outdoor*. Proses pemesanan makanan dan minuman pada Toko Mansure masih menggunakan sistem manual, dimana pelanggan datang langsung menemui bagian kasir untuk melihat menu, melakukan pemesanan makanan dan minuman kemudian melakukan pembayaran dan kasir akan mencetak 2 setruk yang dimana satu untuk pelanggan dan satu untuk bagian koki untuk memproses pesanan pelanggan. Sementara itu, jumlah pelanggan yang datang semakin malam semakin ramai, sehingga menimbulkan beberapa masalah terjadi, seperti antrian pelanggan yang membuat ruangan menjadi penuh yang akan mengakibatkan susah akses jalan untuk karyawan.

Daftar menu yang disediakan belum adanya informasi yang lengkap seperti deskripsi menu mengenai makanan atau minuman sehingga pelanggan harus

bertanya terlebih dahulu dan menyebabkan semakin lamanya antrian. Jika ada penambahan menu baru ataupun perubahan menu yang sudah ada di daftar menu, daftar menu harus diganti ulang, dimana jika hal tersebut dilakukan terus menerus maka akan menambah biaya pengeluaran kafe. Pada saat proses pemesanan yang terjadi, belum adanya informasi mengenai estimasi total pembayaran yang bisa dilihat langsung oleh pelanggan, sehingga membuat pelanggan menghitung terlebih dahulu untuk menyesuaikan dengan *budget* mereka.

Sistem yang berjalan saat ini sangat memungkinkan untuk diubah dengan memanfaatkan teknologi sistem informasi, sehingga dapat meningkatkan kualitas pelayanan pada Toko Mansure kepada pelanggan. Pemanfaatan teknologi informasi berbasis *website* yang dapat diakses oleh pelanggan dan pegawai dianggap lebih baik karena pelanggan tidak perlu menunggu atau mengantri untuk melakukan pemesanan serta dapat mengurangi biaya pengeluaran dalam penggunaan kertas yang digunakan untuk setruk dan daftar menu. Pelanggan dapat memilih sekaligus melakukan pemesanan melalui *gadget* masing-masing dengan mengakses *website* pemesanan makanan dan minuman pada Toko Mansure serta pelanggan dapat melihat langsung estimasi total pembayaran pada sistem. Selain itu, sistem informasi ini dapat memudahkan karyawan khususnya bagian koki dan bagian kasir karena konsep pada penelitian ini ialah pelanggan akan melakukan pemesanan di tempat duduk kemudian data pesanan akan diteruskan dan diproses ke sistem bagian koki selanjutnya jika pesanan sudah selesai, data pesanan akan diteruskan ke sistem bagian kasir dan dilakukan pembayaran oleh pelanggan. Oleh karena itu, bagian kasir tidak perlu lagi memberikan setruk pembayaran ke bagian koki untuk memproses pesanan pelanggan.

Pada penelitian terdahulu yang dilakukan oleh Setiyani pada tahun 2019, dalam penelitiannya dibuatnya aplikasi berbasis *website* yang dapat memberikan kemudahan kepada *user* untuk melakukan pemesanan makanan dan aplikasi hanya bisa diakses dan digunakan di Ramen Shinju. Namun, masih disarankan untuk melakukan pengembangan dalam hal pengelolaan pesanan yang masuk dari pelanggan diproses oleh koki dan penambahan fitur cetak laporan data transaksi pembayaran yang dapat dilakukan oleh bagian kasir.

Berdasarkan permasalahan diatas, perlu adanya sistem informasi pemesanan menu makanan dan minuman untuk memudahkan pelanggan dalam melakukan proses pemesanan makanan dan minuman pada meja masing-masing, membantu mengurangi biaya yang dikeluarkan oleh Toko Mansure dalam mencetak daftar menu dan setruk pembayaran, serta memudahkan karyawan bagian koki dan kasir dalam proses pemesanan makanan, minuman juga pembayaran yang dilakukan oleh pelanggan.

## **1.2. Rumusan Masalah**

Berdasarkan latar belakang yang telah dikemukakan, maka dirumuskan permasalahan pada penelitian ini adalah sebagai berikut.

1. Bagaimana mengimplementasikan aplikasi berbasis *website* yang data pesanan pelanggan dapat diakses oleh bagian koki dan kasir pada Toko Mansure.
2. Bagaimana mengimplementasikan aplikasi berbasis *website* yang mempermudah pelanggan dalam melakukan pemesanan makanan dan minuman, mengetahui detail informasi menu, mempermudah dalam pembaharuan menu.
3. Bagaimana membangun aplikasi berbasis *website* yang dapat meningkatkan kualitas pelayanan pada Toko Mansure.

## **1.3. Batasan Masalah**

Adapun beberapa batasan masalah dalam penelitian ini adalah sebagai berikut.

1. Sistem informasi yang dikembangkan berbasis *website* menggunakan *framework CodeIgniter* 4 dan *bootstrap* 5 dengan bahasa pemrograman PHP, HTML dan CSS.
2. Fokus penelitian ini bagian *front-end* dalam pengembangan sistem informasi pemesanan menu makanan dan minuman.
3. Sistem informasi berbasis *website* ini hanya dapat diakses pelanggan, bagian koki, bagian kasir dan super admin Toko Mansure.

4. Sistem informasi berbasis *website* ini dirancang sesuai dengan kebutuhan dari pihak Toko Mansure.

#### **1.4. Tujuan Penelitian**

Adapun tujuan dilakukannya penelitian ini adalah mengembangkan dan mengimplementasikan *front-end* digitalisasi pemesanan menu makanan dan minuman berbasis *website* pada Toko Mansure.

#### **1.5. Manfaat Penelitian**

Adapun manfaat dari penelitian ini, yaitu:

- a. Bagi pelanggan Toko Mansure
  1. Memudahkan dalam melakukan pemesanan makanan dan minuman.
  2. Memudahkan pelanggan dalam mengetahui informasi mengenai menu makanan dan minuman.
  3. Memudahkan pelanggan dalam mengetahui total jumlah pesanan yang telah dipesan.
- b. Bagi Toko Mansure
  1. Mempermudah admin dalam melakukan pembaharuan informasi menu.
  2. Mengurangi kesalahan dalam pencatatan pesanan pelanggan.
  3. Memudahkan dalam pengelolaan administrasi keuangan.
  4. Meningkatkan daya saing di dunia perbisnisan.

## II. TINJAUAN PUSTAKA

### 2.1. Penelitian Terdahulu

Beberapa penelitian terdahulu yang digunakan sebagai referensi untuk penelitian ini.

#### a) Aplikasi Pemesanan Makanan Berbasis Web Di Ramen Shinju

Ramen Shinju adalah salah satu outlet ramen yang berada di Bandung. Ramen shinju memiliki pengunjung yang ramai sehingga pengunjung rela antri untuk menikmati ramen shinju. Metode pelayanan yang digunakan di ramen shinju yaitu dengan melakukan pencatatan di kertas-kertas pelayan mereka. Hal tersebut memiliki kemungkinan kesalahan pada pendataan pesanan pengunjung dan mengakibatkan ketidaktepatan waktu pesanan datang, karena urutan kertas-kertas yang tidak sengaja tercecer oleh pelayan. Oleh karena itu dibuat aplikasi pemesanan menu makanan berbasis website sehingga dapat mengurangi penggunaan kertas dan memberi informasi mengenai makanan serta minuman di Ramen Shinju.

Metode pengembangan yang digunakan pada penelitian ini yaitu metode *Waterfall*. Proses yang dilakukan oleh pelanggan pada sistem ini adalah pelanggan mengisi data pada sistem atau daftar. Kemudian pelanggan dapat melihat, memilih dan memesan menu yang ada pada Ramen Shinju. Kemudian pesanan yang masuk akan diterima dan diproses oleh admin. Saat pesanan datang dan selesai dihidangkan, pelanggan melakukan pembayaran dibagian kasir.

Penelitian ini berhasil dibuat dengan memberikan kemudahan kepada *user* saat melakukan pemesanan dan hanya bisa diakses di Ramen Shinju. Tetapi, Penelitian ini memerlukan pengembangan dalam hal pengelolaan pesanan yang masuk dari pelanggan akan diproses oleh bagian dapur dan penambahan fitur cetak laporan data transaksi pembayaran yang dilakukan oleh bagian kasir (Setiyani, 2019).



### **b) Perancangan Sistem Aplikasi Pemesanan Makanan dan Minuman pada Cafeteria No Caffe di Tanjung Balai Karimun Menggunakan Bahasa Pemrograman PHP dan MySQL**

Cafeteria No Caffe merupakan salah satu usaha mikro kecil yang menjual makanan dan minuman. Dalam penjualannya Cafeteria No Caffe masih mengalami permasalahan dimana proses pengolahan data penjualan masih dilakukan secara manual, pencatatan pesanan tidak teratur sehingga kurang terjamin dalam hal keamanan data, juga mengalami kesalahan pengantaran pesanan disaat banyaknya pelanggan yang datang. Oleh karena itu, penelitian ini membahas dan bertujuan untuk merancang sistem aplikasi transaksi pemesanan pada Cafeteria No Caffe sehingga dapat memecahkan permasalahan yang ada di Cafeteria No Caffe.

Metode pengembangan yang digunakan pada penelitian ini yaitu metode *Prototype*. Gambaran umum sistem yang diusulkan ini digambarkan sebagai sebuah bentuk fasilitas yang menyediakan informasi bagi konsumen dalam mendapatkan data yang lebih detail tentang menu apa saja yang ditawarkan oleh Cafeteria No Caffe dan dapat melakukan transaksi pemesanan secara langsung. Hasil dari penelitian ini adalah adanya perancangan sistem aplikasi pemesanan makanan dan minuman pada Cafeteria No Caffe di Tanjung Balai Karimun Menggunakan Bahasa Pemrograman PHP dan MYSQL, dengan 4 *user* yaitu konsumen, kasir, bag produksi dan *waiter*.

Namun pada usulan rancangan ini belum terdapat fitur untuk mencetak laporan yang mana dapat memudahkan kasir dalam melakukan pengelolaan laporan data transaksi pembayaran (Kurniawan & Syarifuddin, 2020).

### **c) Rancang Bangun Sistem Informasi Pemesanan Menu Makanan Berbasis Web (Studi Kasus: Restoran Bukit Randu Bandara)**

Restoran Bukit Randu Bandara melayani makanan dari seluruh *customer* yang berada di Tangerang dan merambah ke lokasi yang sangat jauh. Proses yang sedang berjalan pada Restoran Bukit Randu Bandara ini masih menerapkan secara manual, baik daftar menu maupun pencatatan pesanan. Pesanan yang sudah dicatat kemudian diberikan ke bagian dapur yang kemudian akan diproses dan pelanggan akan melakukan pembayaran pada kasir.

Metode pengembangan pada penelitian ini menggunakan metode *waterfall*. Terdapat 3 pengguna dalam aplikasi ini yaitu *customer*, *supervisor* dan admin *user*. Untuk melakukan pemesanan *customer* diwajibkan untuk *login* terlebih dahulu sedangkan *supervisor* melakukan pengelolaan *user* dan pengelolaan laporan. Untuk admin *user* dapat mengelola produk, transaksi dan mencetak data. Kemudian data yang sudah dicetak akan diberikan kepada *customer*.

Meskipun penelitian ini menjadi salah satu solusi bagi customer yang ingin melakukan pemesanan makanan tanpa harus mengantri, penelitian ini memerlukan pengembangan dalam hal pembagian pengguna sistem seperti pengelolaan transaksi dan cetak data sebaiknya dilakukan oleh bagian kasir dan ditambahkan bagian koki untuk memproses pesanan yang masuk dari *customer* (Handayani et al., 2020).

#### **d) Sistem Informasi Penjualan Makanan Berbasis Web pada Rumah Makan Jagosore**

Rumah Makan Jagosore merupakan suatu usaha menengah milik perorangan di Boyolali yang menyediakan masakan soto dan kare. Sebelumnya sistem penjualan pada RM Jagosore masih menggunakan cara konvensional yaitu mencatat nota penjualan dengan bolpoin dan kertas serta menggunakan kalkulator. Laporan penjualan yang didapat dari nota penjualan akan ditulis pada MS. Excel dimana hal tersebut memakan waktu yang lama dalam penulisan. Oleh karena itu pada penelitian ini dibuat suatu sistem informasi penjualan berbasis web untuk memudahkan pembeli dan penjual dalam hal pemesanan makanan, transaksi dan dokumentasi.

Metode pengembangan yang digunakan pada penelitian ini menggunakan metode *waterfall* dengan 2 pengguna sistem, yaitu admin dan kasir. Untuk mengakses *website*, pengguna diharuskan untuk melakukan login terlebih dahulu. Admin dapat mengakses semua fungsi di dalam sistem seperti halaman penjualan, operator, laporan dan data penjualan berupa halaman makanan dan kategori makanan. Sedangkan kasir hanya dapat mengakses halaman penjualan makanan berupa halaman data penjualan dan halaman penjualan.

Sistem sudah diuji dan dapat diterima oleh pengguna di RM Jagosore dengan persentase 77,3% atau skor “Baik”, tetapi penelitian ini memerlukan pengembangan dalam hal pemesanan makanan yang dilakukan langsung oleh pelanggan (Azzis, 2022).

## **2.2. Uraian Landasan Teori**

Beberapa teori yang berkaitan dengan penelitian ini adalah sebagai berikut.

### **2.2.1. Pengertian Sistem**

Menurut (Fatta, 2007) Sistem merupakan kumpulan objek-objek yang saling berhubungan dan saling berinteraksi yang dimana hubungan antar tersebut dapat dianggap sebagai satu kesatuan yang direncanakan untuk mencapai satu tujuan. Sedangkan Murdick dan Ross (1993) memaparkan bahwa sistem adalah kombinasi dari elemen-elemen yang digabungkan untuk mencapai suatu tujuan bersama (Fatta, 2007). Sehingga dapat disimpulkan bahwa sistem terdiri dari elemen-elemen yang digabungkan dan membentuk sebagai satu kesatuan untuk mencapai satu tujuan.

### **2.2.2. Pengertian Informasi**

Informasi adalah kumpulan data yang diproses sedemikian rupa sehingga membuatnya memiliki makna bagi penerima (Anggraeni & Irviani, 2017). Menurut Gordon B. Davis informasi adalah hasil pengolahan data menjadi bentuk yang memiliki arti penting bagi penerima, serta memiliki nilai yang nyata atau dapat dirasakan dalam pengambilan keputusan saat ini maupun yang akan datang (Hutahaean, 2014). Dari pengertian-pengertian tersebut maka secara garis besar informasi adalah kumpulan data atau fakta yang diolah sehingga memiliki manfaat bagi penerimanya dan dapat membantu keraguan saat pengambilan keputusan saat ini ataupun yang akan datang.

### 2.2.3. Pengertian Sistem Informasi

Menurut Husein dan Wibowo (2006), sistem informasi adalah seperangkat komponen yang saling berhubungan yang dapat mengumpulkan, memproses, menyimpan, dan berbagi informasi untuk mendukung pengambilan keputusan dan pengendalian dalam sebuah organisasi (Pribadi et al., 2022). Sedangkan menurut (Anggraeni & Irviani, 2017) sistem informasi merupakan suatu penggabungan yang teratur antara individu, perangkat keras (*hardware*), perangkat lunak (*software*), jaringan komunikasi dan sumber daya data yang bekerja sama untuk dapat mengumpulkan, mengubah dan menyebarkan informasi di dalam sebuah organisasi. Maka dapat disimpulkan bahwa sistem informasi terdiri dari komponen-komponen yang teratur, diantaranya individu, *hardware*, *software*, jaringan komunikasi dan sumber daya data yang berfungsi untuk mengumpulkan, mengolah, menyimpan dan mendistribusikan untuk pengambilan keputusan dan pengendalian organisasi.

### 2.2.4. Pengertian Digitalisasi

Digitalisasi adalah proses perubahan media dari format cetak, audio atau video menjadi format digital (Asaniyah, 2017). Sedangkan menurut kamus istilah Gartner.com, digitalisasi adalah memperbaharui sebuah model bisnis dan menyediakan pendapatan-pendapatan serta peluang-peluang nilai yang menghasilkan dalam penggunaan teknologi digital (Syahra et al., 2022).

### 2.2.5. Pengertian Website

*Website* adalah kumpulan halaman-halaman web yang terdapat di dalam suatu domain dan berisi informasi. Biasanya, sebuah website terdiri dari banyak halaman web yang saling terhubung. Hubungan antara halaman web satu dengan web yang lainnya disebut *hyperlink*, sementara teks yang berfungsi sebagai media penghubung disebut *hypertext* (Yuhefizar et al., 2009). Menurut Hakim Lukmanul,

*Website* adalah suatu sarana internet yang berfungsi sebagai penghubung antara dokumen-dokumen dalam skala lokal maupun global. Dokumen-dokumen ini disebut *web page*. Sementara *link* dalam *website* pengguna dapat berpindah dari satu halaman ke halaman lainnya (*hypertext*), baik itu di dalam server yang sama maupun *server* di seluruh dunia. Halaman-halaman ini dapat diakses melalui perangkat lunak *browser* seperti *Internet Explorer*, *Mozilla Firefox*, *Google Chrome* atau aplikasi *browser* lainnya (Iqbal, 2022). Maka disimpulkan *website* adalah halaman web atau *web page* dalam skala lokal yang berisi informasi yang dihubungkan dengan *web page* dalam skala global.

#### **2.2.6. Pengertian HTML**

Sebuah bahasa *formatting* yang berguna untuk membuat situs halaman *website* disebut HTML (*Hyper Text Markup Language*). Dalam dunia pemrograman *website* (Web Programming), HTML menjadi pondasi fundamental pada halaman *website*. HTML memiliki beberapa elemen yang tersusun dari tag-tag yang memiliki fungsi nya masing-masing (Fauzi & Ismatullah, 2022). Sedangkan menurut (Aprilia, 2021) HTML (*Hyper Text Markup Language*) adalah bahasa *markup* yang digunakan untuk membuat halaman *website* yang terdiri dari kombinasi teks dan simbol yang disimpan dalam sebuah *file*. Saat membuat *file* HTML, terdapat format khusus yang harus diikuti yang dimana telah tertuang dalam standar kode internasional atau biasa disebut dengan ASCII (*American Standard Code for Information Interchange*). *File* HTML dapat diakses melalui aplikasi *browser* seperti *Internet Explorer*, *Chrome*, *Mozilla Firefox* atau aplikasi *browser* lainnya. Maka disimpulkan bahwa HTML adalah sebuah file yang berisi bahasa *markup* yang berfungsi untuk membuat halaman *website*. HTML disusun dari tag-tag yang memiliki fungsi masing-masing sesuai dengan standar kode internasional (ASCII).

### 2.2.7. Pengertian CSS

CSS (*Cascading Style Sheets*) pada dasarnya adalah “bahasa *style sheet*” yang berarti Bahasa pemrograman untuk mendesain sebuah *website*. CSS adalah bahasa pemrograman yang digunakan untuk mendesain halaman *website*. Saat mendesain halaman *website*, CSS menggunakan penanda yang dikenal sebagai *id* dan *class*. CSS dapat *mengubah font*, ukuran, warna dan format *font*, mengatur ukuran layout, lebar, tinggi, dan warna *element*, mengubah tampilan *form*, membuat halaman *website* yang *responsive* dan lainnya (Fauzi & Ismatullah, 2022). HTML dan CSS memiliki kaitan, HTML merupakan pondasi saat membuat halaman *website* sedangkan CSS untuk membenahi gaya nya dengan berbagai macam *syntax*.

### 2.2.8. Pengertian CodeIgniter

CodeIgniter adalah sebuah kerangka kerja aplikasi web (*Web Application Framework/WAF*) yang diciptakan khusus untuk mempermudah pengembang web dalam mengembangkan aplikasi berbasis *website*. CodeIgniter menyediakan kumpulan kode berupa pustaka (*library*) dan alat (*tools*) yang digabungkan menjadi suatu kerangka kerja (*framework*) yang terintegrasi. CodeIgniter mengikuti pola desain atau arsitektur *Model-View-Controller* (MVC) yang memisahkan bagian logika bisnis dengan tampilan. Dengan menggunakan pola desain ini, pengembang web dapat bekerja secara bersama-sama dalam mengembangkan aplikasi berbasis web, dengan fokus pada tugas masing-masing bagian tanpa mengganggu bagian yang lainnya. Hal ini memungkinkan aplikasi yang dibangun dapat diselesaikan dengan lebih cepat dan efisien (Asyikin, 2019). Gambaran komponen MVC adalah sebagai berikut (Asyikin, 2019).

1. *Model*

*Model* menggambarkan data yang digunakan pada aplikasi, seperti data dari database, RSS atau hasil pemanggilan API serta operasi *Create, Read, Update* dan *Delete* yang terlibat dalam pengelolaan data.

2. *View*

*View* merupakan informasi yang ditampilkan kepada pengguna melalui *browser*. Umumnya, *view* berupa file HTML atau kode PHP yang membentuk *template* untuk sebuah situs *website*. Pada CodeIgniter, *view* dapat berupa bagian-bagian dari halaman, *template* atau jenis lain halaman atau *template* lainnya.

### 3. *Controller*

Berperan sebagai logika bisnis yang menjadi penghubung antara *model* dan *view*. *Controller* akan merespons permintaan HTTP yang diterima dari pengguna (melalui *browser*). Dari permintaan ini, *controller* akan menentukan tindakan apa yang harus dilakukan.






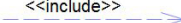
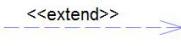
#### 2.2.9. Pengertian *Unified Modeling Language (UML)*

UML (*Unified Modeling Language*) adalah suatu ‘bahasa’ pemodelan yang digunakan untuk sistem atau perangkat lunak yang berbasis paradigma ‘*Object Oriented*’. Pemodelan (*modeling*) sebenarnya digunakan untuk menyederhanakan permasalahan yang kompleks agar lebih mudah dipelajari dan dipahami. Tujuan dari pemodelan (dalam konteks pengembangan sistem atau aplikasi perangkat lunak aplikasi) adalah untuk melakukan analisis, memahami, memvisualisasikan dan berkomunikasi antar anggota tim pengembang. Selain itu, pemodelan juga berfungsi sebagai sarana dokumentasi yang berguna untuk mempelajari perilaku perangkat lunak secara rinci dan melakukan pengujian terhadap perangkat lunak yang telah selesai dikembangkan (Nugroho, 2010). Adapun beberapa diagram yang dideskripsikan oleh UML adalah sebagai berikut.

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

*Use case diagram* menunjukkan sebuah interaksi yang dilakukan oleh aktor dan proses yang dilakukan oleh sistem terhadap sistem informasi yang akan dibuat (Sari, 2021). Adapun simbol-simbol *use case diagram* disajikan pada Tabel 1.


Tabel 1. Simbol-Simbol *Use Case Diagram*

No.	Gambar	Nama	Keterangan
1		<i>Actor</i>	aktor sebagai peran yang berinteraksi dengan sistem.
2		<i>Dependency</i>	Relasi yang menghubungkan perubahan salah satu elemen yang berpengaruh pada elemen lain.
3		<i>Generalization</i>	Relasi yang menghubungkan objek anak dan objek induk, dimana objek anak mewarisi sifat dan method dari induk.
4		<i>Association</i>	Relasi yang menghubungkan antara satu objek dengan objek lainnya.
5		<i>Use Case</i>	Menggambarkan bagaimana aktor menggunakan sistem.
6		Relasi <i>Include</i>	Relasi yang menghubungkan suatu <i>use case</i> dengan <i>use case</i> tambahan, dimana <i>use case</i> tambahan tidak dapat menjalankan fungsinya tanpa <i>use case</i> .
7		Relasi <i>Extend</i>	Relasi yang menghubungkan suatu <i>use case</i> dengan <i>use case</i> tambahan, dimana <i>use case</i> tambahan dapat menjalankan fungsinya walau tanpa <i>use case</i> .


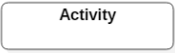



## 2. *Activity Diagram*

*Activity diagram* menunjukkan alur setiap aktivitas aktor pada suatu sistem. *Activity diagram* adalah Teknik untuk menggambarkan logika prosedural, proses bisnis, dan jaringan kerja antara pengguna dan sistem. *Activity diagram* dibuat berdasarkan sebuah atau beberapa *use case* pada *use case diagram* (Sari, 2021). Adapun simbol-simbol *activity diagram* disajikan pada Tabel 2.

Tabel 2. Simbol-Simbol *Activity Diagram*

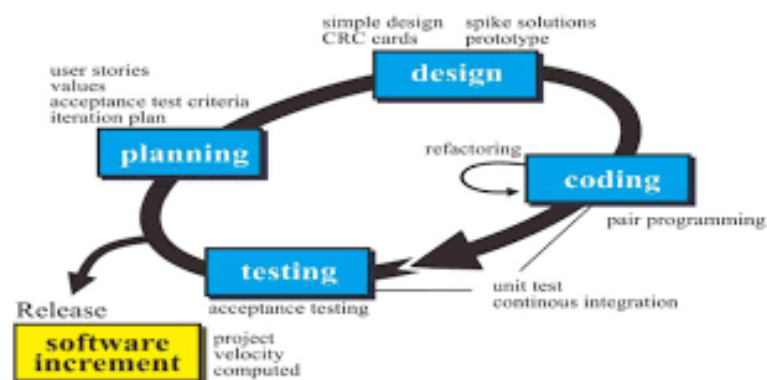
No.	Gambar	Nama	Keterangan
1		<i>Initial State</i>	Tanda awal dimulainya aktivitas yang dilakukan sistem.



No.	Gambar	Nama	Keterangan
2		<i>Final Node</i>	Tanda akhir selesainya aktivitas yang dilakukan sistem.
3		<i>Activity</i>	<i>Symbol</i> aktivitas atau yang sedang dilakukan dalam sistem dan diawali dengan kata kerja.
4		<i>Decision</i>	<i>Symbol</i> percabangan dimana terdapat pilihan yang lebih dari satu.
5		<i>Join</i>	<i>Symbol</i> penggabungan <i>activity</i> dengan aksi paralel.
6		<i>Swimlane</i>	<i>Diagram flow</i> yang menggambarkan objek melakukan aktivitas.

### 2.2.10. Pengertian XP (*Extreme Programming*)

*Extreme Programming* (XP) adalah salah satu metodologi rekayasa perangkat lunak yang banyak digunakan oleh para pengembang untuk mengembangkan aplikasi. XP sangat cocok untuk proyek pengembangan yang membutuhkan kemampuan adaptasi yang cepat terhadap perubahan yang terjadi selama proses pengembangan aplikasi. XP juga cocok digunakan ketika tim pengembang terdiri dari jumlah anggota yang tidak terlalu banyak dan berada pada lokasi yang sama selama pengembangan sistem (Suryantara, 2017). Pengembangan perangkat lunak menggunakan XP terdiri dari beberapa tahapan, diantaranya sebagai berikut (Suryantara, 2017).



Gambar 1. Metodologi XP.

Penjelasan tahapan-tahapan metode *extreme programming* adalah sebagai berikut.

1. *Planning*/Perencanaan

Tahap perencanaan dimulai dengan pemahaman konteks bisnis dari aplikasi, mendefinisikan hasil yang diinginkan (*output*), fitur-fitur yang akan ada dalam aplikasi, fungsi dari aplikasi yang sedang dikembangkan, menentukan jadwal dan anggaran pengembangan aplikasi serta merancang alur pengembangan aplikasi.

2. *Design*/Perancangan

Tahap perancangan fokus pada perancangan atau desain aplikasi dengan cara yang sederhana.

3. *Coding*/Pengkodean

Tahap pengkodean melibatkan aktivitas utama dalam pengembangan aplikasi menggunakan XP yaitu *pair programming* (melibatkan dua atau lebih programmer dalam membuat program)

4. *Testing*/Pengujian

Tahap pengujian berfokus pada pengujian fitur-fitur yang ada dalam aplikasi untuk memastikan bahwa tidak ada kesalahan (*error*) dan aplikasi yang dibuat sesuai dengan proses bisnis yang diinginkan oleh klien (pelanggan).

### **2.2.11. Pengertian *Front-End***

*Front-end* adalah bagian dari sistem yang bertugas memberikan pengguna antarmuka visual. Fungsinya adalah untuk mengembangkan komponen-komponen visual dalam sistem dan bertanggung jawab atas penampilan antarmuka (Rizaldy & Dirgahayu, 2020). Seorang *Front-end* adalah individu yang memiliki tanggung jawab dalam mengembangkan tampilan dari sebuah situs *website* ataupun aplikasi menggunakan bahasa pemrograman HTML, CSS dan JavaScript (Dharma, 2021). Intinya, *front-end developer* ialah menciptakan tampilan aplikasi yang mudah digunakan dan *user friendly*, bukan hanya sekedar desain yang menarik secara visual (Mobile, 2023).

*Front end developer* umumnya bekerja secara kolaboratif dengan *back end developer* dan keduanya merupakan bagian dalam proses pengembangan aplikasi. Perbedaannya, *front end developer* memiliki tanggung jawab terhadap tampilan dari sebuah aplikasi atau *website*, sementara *back end developer* berfokus pada bagian yang tidak terlihat secara langsung, seperti fitur dan perintah yang ada dalam aplikasi tersebut (Mobile, 2023). Secara umum, kriteria *front-end* yang baik menurut (Yasin, 2019) adalah sebagai berikut.

1. *Responsiveness*

*Responsive web design* adalah konsep desain *website* yang dapat menyesuaikan tampilan dengan berbagai ukuran tiap *device*. Manfaat dari *responsive web design* adalah kemampuan sebuah *website* untuk beradaptasi pada tata letak ukuran tiap *device* dengan ukuran *font*, gambar dan komponen lainnya tanpa harus melakukan *horizontal scrolling* (Novianty, 2017).

2. Kemudahan penggunaan

*Front-end* yang baik harus mudah digunakan dan dapat memberikan pengalaman pengguna yang baik. Salah satu bidang ilmu untuk menganalisa atau menguji tingkat kemudahan penggunaan perangkat lunak ialah *usability* (Ependi et al., 2019).

3. Kecepatan

Salah satu indikator *website* yang baik adalah dapat diakses dengan cepat. *Website* yang dapat diakses dengan lambat dan cenderung berat memberikan kesan yang kurang menarik. Oleh karena itu, perlu untuk memastikan *website* dapat diakses dengan cepat dan ringan (Yasin, 2019).

4. Aksesibilitas

Adapun tujuan utama dari aksesibilitas *website* (*website accessibility*) adalah memastikan bahwa membuat halaman pada *website* lebih mudah untuk dinavigasi dan dibaca untuk setiap pengguna yang mengunjungi situs, terlepas dari berbagai kondisi pengguna (sebagai contoh difabel, orang tua, dan kondisi lainnya yang bisa menyulitkan untuk mengakses *website* biasa) (Ismail & Kuppusamy, 2019). *Tools* evaluasi aksesibilitas yang digunakan adalah aXe DevTools dari Deque (Hafiar et al., 2022).

### 2.2.12. Pengertian *Responsively App*

*Responsively App* adalah *open source dev tool* dan gratis untuk pengembangan web *responsive*. Aplikasi ini tersedia untuk Linux, Microsoft Windows dan MacOS. Ini adalah browser yang dimodifikasi yang menggunakan Electron, yang menampilkan aplikasi web di beberapa perangkat secara bersamaan dan dalam satu jendela dengan interaksi pengguna yang dicerminkan, DevTools, dan lainnya (Logix, 2021). Untuk menggunakan *responsively app* hanya cukup memasukkan URL pada *field* yang tersedia. *Responsively App* juga dapat mengatur *device* pada fitur “*Manage Suites*”.

### 2.2.13. Pengertian *Usability* (Kegunaan)

Menurut (Patria, 2021) dalam artikelnya, salah satu tugas dari seorang *front-end developer* yaitu bekerja sama dengan *back-end developer* dan *web designer* untuk melakukan *usability testing* (pengujian kegunaan program) dan meningkatkan kebergunaannya. *Usability* atau “ketergunaan” merujuk pada tingkat kualitas suatu sistem yang mudah dipelajari, mudah digunakan dan mendorong pengguna untuk menggunakan sistem sebagai alat bantu positif dalam menyelesaikan tugas. Dalam konteks ini, sistem yang dimaksud adalah perangkat lunak. *Usability* juga dapat diartikan sebagai suatu ukuran dimana pengguna dapat mengakses fungsionalitas sistem dengan efektif, efisien dan memuaskan untuk mencapai tujuan tertentu (Handiwidjojo & Ernawati, 2016).

### 2.2.14. Pengertian *Usability Testing*

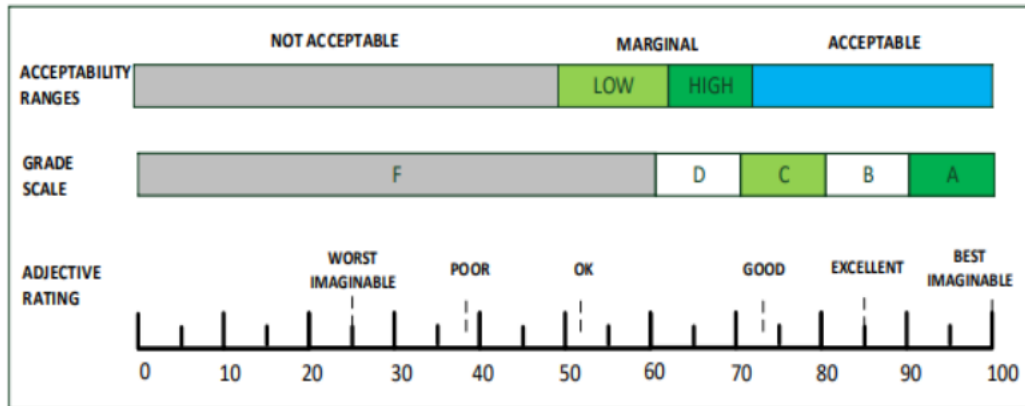
*Usability testing* merupakan salah satu cara untuk mengetahui apakah *user* dapat mudah menggunakan aplikasi, seberapa efisien dan efektif sebuah aplikasi dapat membantu *user* dalam mencapai tujuannya dan apakah *user* puas dengan aplikasi yang digunakan. Untuk melakukan *usability testing* dapat menggunakan *wireframe* dari aplikasi ataupun aplikasi yang sudah dibuat. *Usability testing*

sebaiknya dilakukan untuk mengetahui terlebih dahulu kemungkinan kendala penggunaan aplikasi yang dihadapi pengguna. *Usability* mengacu pada kualitas pengalaman *user* pada saat menggunakan aplikasi baik berupa itu peralatan ataupun aplikasi berbasis *website*, *desktop* atau *mobile* (Wahyudi et al., 2021). Menurut Liu (2008), *usability testing* didefinisikan sebagai sebuah tahapan penelitian dengan memberikan serangkaian *task* yang terkait dengan antarmuka kepada pengguna asli suatu sistem (Isnainrajab et al., 2020).

### **2.2.15. Pengertian System Usability Scale**

Salah satu metode dalam melakukan analisa atau pengujian *usability* adalah SUS, dengan melibatkan pengguna akhir atau *end user* dalam proses pengerjaannya (Fatmawati, 2021). SUS adalah alat ukur untuk menilai *usability* yang dapat diandalkan dan valid. Metode ini memiliki performa yang baik, bahkan lebih baik daripada kuesioner komersial maupun kuesioner yang dikembangkan secara internal. Dapat diandalkan yang dimaksud pada metode ini ialah merujuk pada konsisten pengguna dalam menanggapi item yang diuji, sedangkan valid yang dimaksud ialah merujuk pada seberapa baik sesuatu dapat mengukur apa yang dimaksudkan untuk diukur. Dalam konteks ini, validitas terkait dengan pengukuran kegunaan yang dirasakan (Sauro, 2011). Metode SUS memiliki beberapa keunggulan, antara lain proses evaluasinya lebih mudah dan dapat dipahami oleh responden, memberikan hasil yang maksimal meskipun dengan jumlah sampel yang sedikit, serta mampu membedakan apakah aplikasi dapat digunakan dengan baik atau tidak (Fatmawati, 2021). SUS memiliki beberapa karakteristik yang membuat menarik dan berbeda dari kuesioner lainnya. Pertama, SUS terdiri dari sepuluh (10) pernyataan, sehingga responden dapat menyelesaikan dengan cepat dan mudah. Kedua, SUS hampir dapat digunakan secara luas untuk mengevaluasi semua jenis antarmuka, termasuk *smartphone*, respon suara interaktif (IVR), sistem (*touch-tone* dan *speech*), TV dan lainnya, karena SUS menggunakan teknologi agnostik. Ketiga, hasil kuesioner SUS berupa nilai tunggal yang berkisar antara 0 hingga 100, sehingga mudah dipahami oleh individu maupun kelompok dari berbagai disiplin (Sidik, 2018). Terdapat tiga aspek utama tingkat penilaian

meliputi *acceptability* (penerimaan), *grade scale* (skala kelas) dan *adjective rating* (peringkat) seperti yang dilihat pada Gambar 2. Kesimpulan diperoleh dengan *grade scale* melalui ketentuan rata-rata:  $A > 80,3$ ,  $74 < B < 80,3$ ,  $68 < C < 74,3$ ,  $51 < D < 68$ , dan  $F < 51$  (Fatmawati, 2021).



Gambar 2. Penentuan Hasil Evaluasi *Usability* dengan SUS.

### 2.2.16. Pengertian *Lighthouse*

*Lighthouse* adalah *tool open-source* gratis yang dapat membantu meningkatkan kecepatan, kinerja dan pengalaman situs web secara keseluruhan (Morris, 2022). Cara kerja *lighthouse* dengan melakukan audit untuk lima kategori pengoptimalan situs website, diantaranya *performance*, *accessibility*, *best practices*, SEO dan *Progressive Web Application*. Dalam kategori *performance*, *lighthouse* mengukur seberapa cepat sebuah situs web dimuat dan seberapa cepat pengguna dapat mengaksesnya (Pol, 2023). *Performance* melibatkan kemampuan *website* untuk memberikan waktu respon yang cepat, sehingga pengguna dapat dengan mudah mengakses informasi yang dibutuhkan (Wisnuadhi et al., 2022). *Lighthouse* memiliki beberapa opsi untuk dapat dijalankan, yaitu melalui *Chrome DevTools*, dari baris perintah atau sebagai modul Node (*Lighthouse*, 2016). Adapun tahapan menggunakan *lighthouse* melalui *Chrome DevTools* menurut (Juviler, 2021) adalah sebagai berikut.

- a. Buka halaman web yang akan diuji.
- b. Buka panel *DevTools* dengan cara klik kanan halaman kemudian pilih *Inspect*. Cara lain membuka *DevTools* dengan memilih **View > Developer >**

**Developer Tools**, atau bisa juga menggunakan keyboard dengan menekan **Ctrl+Shift+C** pada Windows atau **Command+Option+C** pada Mac.

- c. Pada panel DevTools, pilih Lighthouse pada menu tab atas. Kemudian klik *Generate Report*. Secara *default*, lighthouse akan melakukan penilaian lima kategori yang disimulasikan pada perangkat. Sebelum melakukan *generate report*, kelima kategori dapat dilakukan untuk diaktifkan atau tidak.

### 2.2.17. Pengertian AXE DevTools

Axe DevTools adalah sebuah ekstensi chrome dan pemeriksa aksesibilitas untuk pengembang, penguji dan desainer (Hristov et al., 2023). Menurut artikel (Gravityforms, 2022), Axe Developer Tools dikembangkan oleh Deque Systems, berdasarkan axe-core. Saat ini adalah alat pemeriksaan aksesibilitas terbaik dan direkomendasikan untuk pengembang dan dapat dengan mudah digunakan selama pengembangan. Adapun tahapan menggunakan Axe DevTools berdasarkan (Deque, n.d.) adalah sebagai berikut.

- a. Install Axe DevTools *Extension* pada chrome atau firefox.
- b. Buka Chrome dan arahkan pada halaman yang akan di uji.
- c. Buka panel DevTools dengan cara klik kanan, lalu pilih “*Inspect*” atau “*Inspect Element*”. Alternatifnya, dapat menggunakan keyboard dengan menekan **Ctrl+Shift+C** pada Windows atau **Command+Option+C** pada Mac.
- d. Pilih Tab “Axe DevTools” pada menu tab atas.
- e. Jalankan pemeriksaan aksesibilitas dengan klik tombol “Scan all of my page”.
- f. Setelah pemeriksaan selesai, akan ditampilkan daftar masalah aksesibilitas yang telah diidentifikasi oleh Axe.
- g. Periksa dan lihat tingkat keparahan dan deskripsi masalah.
- h. Ikuti panduan penyelesaian masalah untuk mempelajari cara memperbaiki masalah.

### **III. METODE PENELITIAN**

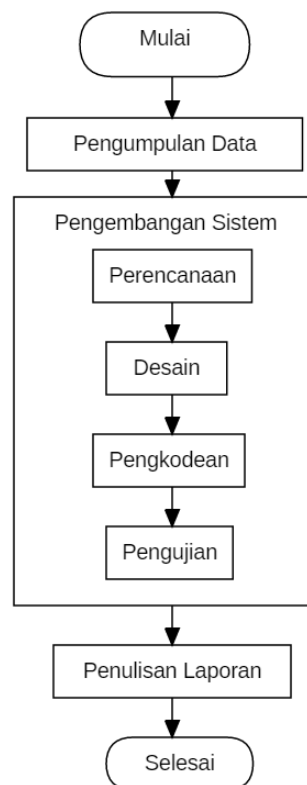
#### **3.1. Waktu dan Tempat Penelitian**

Penelitian ini dilakukan di Jurusan Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Lampung, yang terletak di Jalan Soemantri Brojonegoro Nomor 1, Gedong Meneng, Bandar Lampung dan Toko Mansure di Jalan KH. Mas Mansur Nomor 74, Rawa Laut, Bandar Lampung.

#### **3.2. Tahapan Penelitian**

Ada tiga tahapan penelitian yang dilakukan dalam penelitian pada Toko Mansure ini yaitu pengumpulan data, pengembangan sistem dan penulisan laporan. Tahapan penelitian dapat dilihat pada Gambar 3.





Gambar 3. Tahapan Penelitian.

### 3.2.1. Tahapan Pengumpulan Data

Pengumpulan data pada penelitian ini adalah sebagai berikut.

a. Observasi

Pada tahapan ini penulis melakukan pengamatan secara langsung pada Toko Mansure.

b. Wawancara

Pada tahapan ini melakukan pertemuan dan tanya jawab langsung kepada Bapak Adit selaku salah satu pemilik Toko Mansure. Proses wawancara dilakukan untuk mendapatkan informasi seperti bagaimana proses bisnis yang sedang berjalan dan kendala yang dihadapi.

c. Studi Pustaka

Pada tahapan ini dilakukan pengumpulan data dan informasi melalui dokumen-dokumen yang bersumber dari jurnal, buku, internet dan lainnya yang berkaitan dengan topik penelitian.

### 3.2.2. Tahapan Pengembangan Sistem

Tahapan pengembangan sistem dalam penelitian ini menggunakan metode *extreme Programming (XP)*. Tahapan-tahapan pengembangan sistem pada metode *Extreme Programming* adalah perencanaan, desain, pengkodean dan pengujian.

#### 3.2.2.1. Perencanaan

Pada tahapan perencanaan ini diawali dengan memahami konteks bisnis dari sistem, menjelaskan keluaran (*output*), fitur-fitur dan fungsi dari aplikasi yang dibuat (Suryantara, 2017). Konteks bisnis dari sistem direpresentasikan dengan kebutuhan fungsional dan non-fungsional.

##### a. Kebutuhan Fungsional

Kebutuhan fungsional adalah kebutuhan yang berkaitan dengan proses-proses yang dapat dilakukan oleh sistem. Kebutuhan fungsional dari sistem ini sebagai berikut.

1. Sistem dapat menampilkan halaman beranda.
2. Sistem dapat menampilkan halaman menu makanan dan minuman.
3. Sistem dapat menambah, mengedit dan menghapus pesanan untuk pelanggan.
4. Sistem dapat menghitung total pembayaran.
5. Sistem dapat menampilkan halaman tagihan pembayaran.
6. Sistem dapat menampilkan halaman dashboard untuk bagian super admin, kasir dan koki.
7. Sistem dapat menambahkan, mengedit dan menghapus daftar menu untuk bagian super admin.
8. Sistem dapat mengubah status pesanan pada koki dan kasir.
9. Sistem dapat melihat dan mencetak laporan keuangan untuk bagian kasir dan super admin.
10. Sistem dapat menginputkan jumlah bayar pelanggan untuk bagian kasir.
11. Sistem dapat mencetak setruk pembayaran untuk bagian kasir.
12. Sistem dapat melihat, menambahkan, menghapus dan mereset kata sandi *user* untuk super admin.

b. Kebutuhan Non-Fungsional

Kebutuhan non-fungsional adalah kebutuhan operasional. Kebutuhan non-fungsional dari sistem ini dibagi menjadi dua, yaitu analisis kebutuhan perangkat lunak (*software*) dan kebutuhan perangkat keras (*hardware*).

1. Analisis Kebutuhan Perangkat Lunak (*Software*)

- a. Sistem Operasi Windows 11 64-bit digunakan untuk menyelesaikan penelitian ini.
- b. *Visual Studio Code* versi 1.74.2 digunakan untuk membuat dan mengedit *source code* dalam mengembangkan sistem informasi pemesanan makanan dan minuman pada Toko Mansure.
- c. Figma digunakan untuk merancang desain *interface website* pemesanan makanan dan minuman pada Toko Mansure.
- d. XAMPP versi 7.4.33 digunakan sebagai server lokal untuk memudahkan dalam menjalankan sistem informasi tanpa menggunakan koneksi internet.
- e. Web browser (Google Chrome) digunakan untuk mencari informasi-informasi yang bersumber dari internet.
- f. *Microsoft Word* digunakan untuk membuat penulisan laporan.
- g. *Star UML* digunakan untuk membuat bahasa pemodelan visual seperti *use case diagram*, *activity diagram* dan *flowchart*.

2. Analisis Kebutuhan Perangkat Keras (*Hardware*)

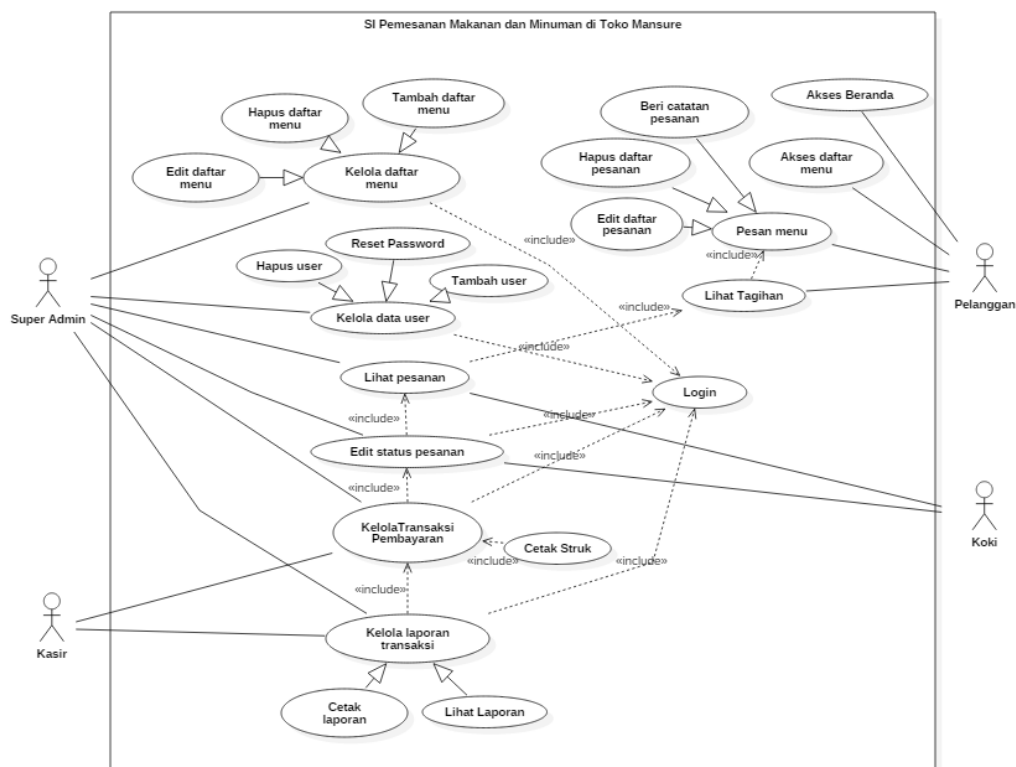
- a. *System Manufacturer* : ASUS
- b. *System Model* : ASUS VivoBook K413EQ
- c. *Processor* : Intel® Core™ i7- 1165G7 CPU @  
2.80GHz
- d. *Installed RAM* : 8 GB
- e. *System Type* : 64 Bit

### 3.2.2.2. Desain

Pada tahapan ini membuat desain yang menggambarkan cara kerja sistem menggunakan UML dan membuat desain *interface* secara sederhana.

#### A. Use Case Diagram

Dalam *use case diagram* ini dijabarkan mengenai interaksi antara *user* dengan sistem itu sendiri, terdapat 4 aktor yang bertindak sebagai *user*, yaitu pelanggan, koki, kasir dan super admin yang akan digambarkan melalui *use case diagram* yang dapat dilihat pada Gambar 4.



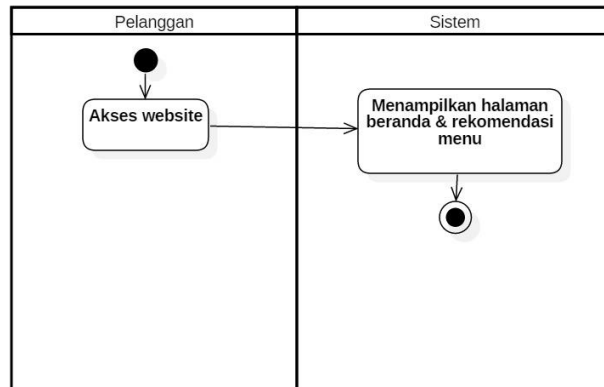
Gambar 4. Use Case Diagram.

#### B. Activity Diagram

Pada *Activity diagram* ini dijabarkan alur setiap proses-proses yang terjadi pada sistem. *Activity diagram* ini adalah pengembangan dari *use case diagram* di atas sehingga memiliki alur aktivitas masing-masing.

### 1) Activity Diagram Akses Beranda

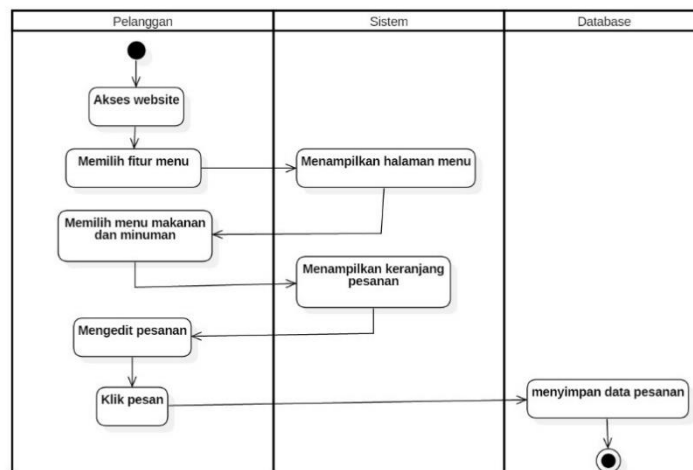
Pelanggan mengakses *website* pertama kali akan langsung menampilkan halaman beranda. Berikut *activity diagram* akses beranda yang dapat dilihat pada Gambar 5.



Gambar 5. Activity Diagram Akses Beranda.

### 2) Activity Diagram Pesan Menu

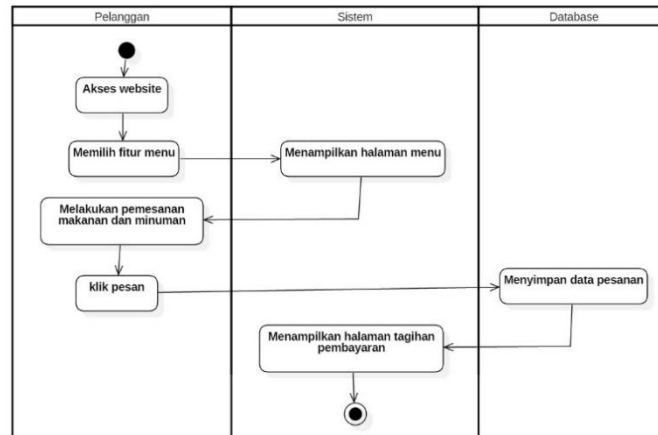
Setelah mengakses halaman beranda, pelanggan dapat langsung memesan makanan dan minuman yang tersedia pada halaman menu dan dapat melihat serta mengedit daftar pesanan yang sudah dipilih pada keranjang pesanan. Berikut *activity diagram* pesan menu yang dapat dilihat pada Gambar 6.



Gambar 6. Activity Diagram Pesan Menu.

### 3) *Activity Diagram* Tagihan Pembayaran

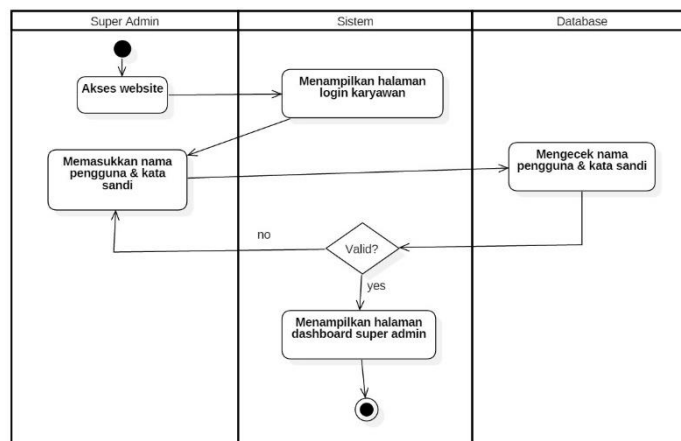
Setelah melakukan pemesanan dan klik pesan, pelanggan dapat melihat tagihan pembayaran. Berikut *activity diagram* tagihan pembayaran yang dapat dilihat pada Gambar 7.



Gambar 7. *Activity Diagram* Tagihan Pembayaran.

### 4) *Activity Diagram* Login Super Admin

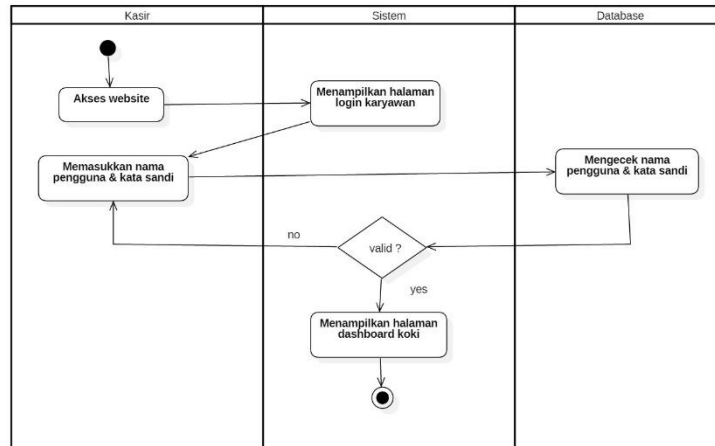
Sebelum masuk ke halaman *dashboard*, super admin diharuskan untuk melakukan *login* terlebih dahulu. Berikut *activity diagram* login super admin yang dapat dilihat pada Gambar 8.



Gambar 8. *Activity Diagram* Login Super Admin.

### 5) Activity Diagram Login Kasir

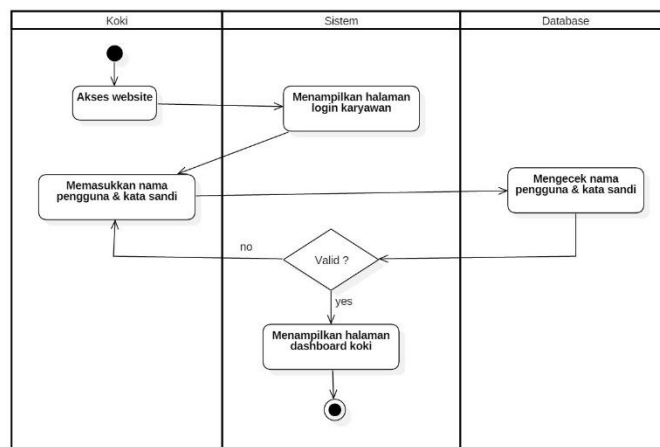
Sebelum masuk ke halaman *dashboard*, kasir yang sudah terdaftar diharuskan untuk melakukan *login* terlebih dahulu. Berikut *activity diagram login* kasir yang dapat dilihat pada Gambar 9.



Gambar 9. Activity Diagram Login Kasir.

### 6) Activity Diagram Login Koki

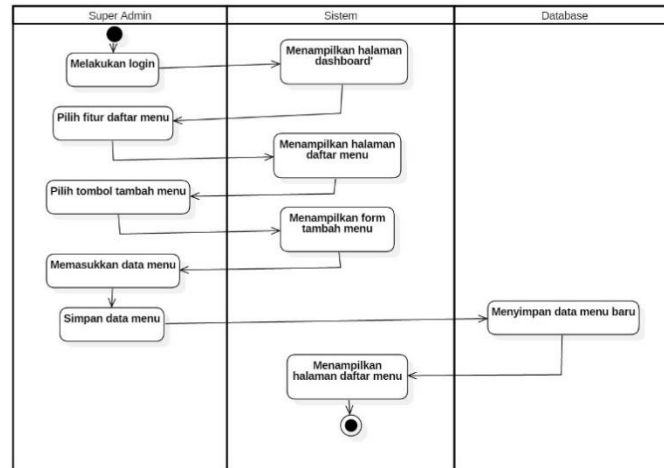
Sebelum masuk ke halaman *dashboard*, koki yang sudah terdaftar diharuskan untuk melakukan *login* terlebih dahulu. Berikut *activity diagram login* koki yang dapat dilihat pada Gambar 10.



Gambar 10. Activity Diagram Login Koki.

### 7) *Activity Diagram* Tambah Menu

Super admin dapat melakukan tambah menu dengan mengakses fitur daftar menu dan mengisi informasi-informasi mengenai menu. Berikut *activity diagram* tambah menu yang dapat dilihat pada Gambar 11.

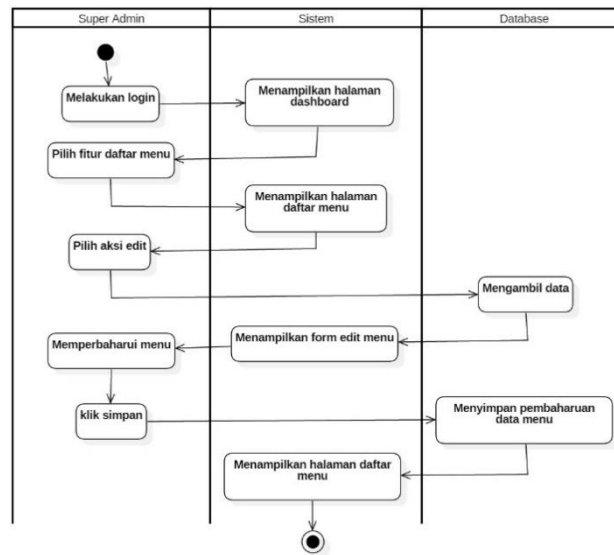


Gambar 11. *Activity Diagram* Tambah Menu.

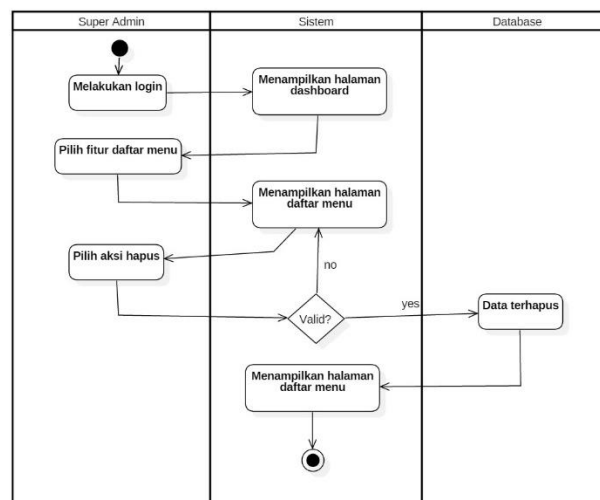
### 8) *Activity Diagram* Edit dan Hapus Menu

Selain dapat menambahkan menu, super admin juga dapat melakukan edit dan juga menghapus menu yang sudah ada. Berikut *activity diagram* edit menu yang dapat dilihat pada Gambar 12 dan *activity diagram* hapus menu yang dapat dilihat pada Gambar 13.





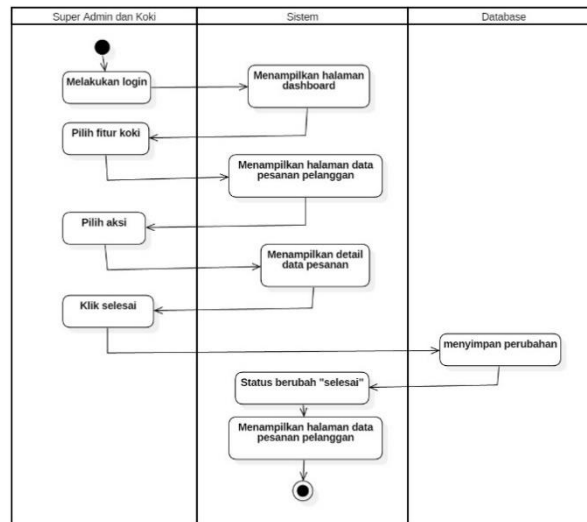
Gambar 12. *Activity Diagram* Edit Menu.



Gambar 13. *Activity Diagram* Hapus Menu.

#### 9) *Activity Diagram* Melihat dan Mengubah Status Pesanan Bagian Koki

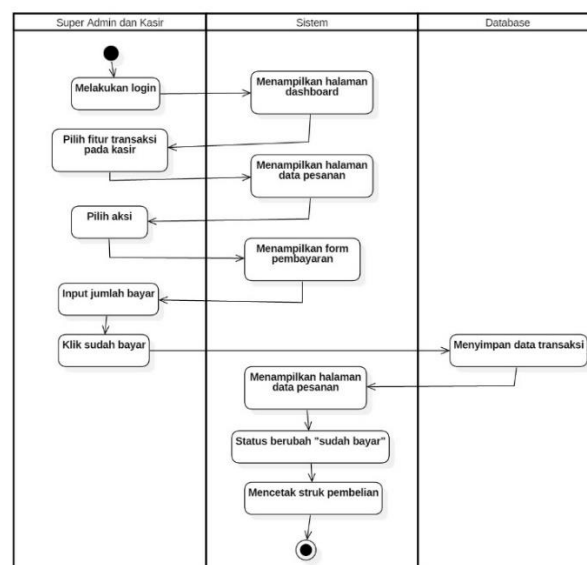
Pesanan yang masuk dari pelanggan akan menuju ke bagian koki, kemudian super admin ataupun koki dapat melihat serta mengubah status pesanan menjadi selesai. Berikut *activity diagram* melihat dan mengubah status pesanan yang dapat dilihat pada Gambar 14.



Gambar 14. *Activity Diagram* Melihat dan Mengubah Status Pesanan.

#### 10) *Activity Diagram* Melihat dan Mengubah Status Pesanan Bagian Kasir

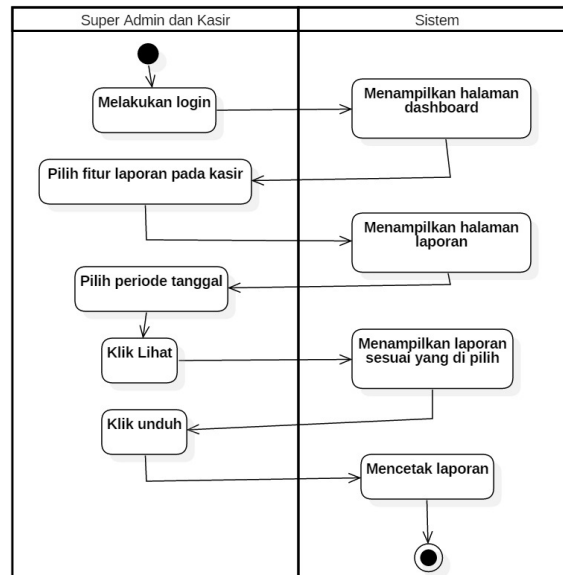
Pesanan yang status nya selesai akan diteruskan ke bagian kasir untuk pelanggan melakukan pembayaran. Kasir dapat menginputkan jumlah bayar pelanggan dan mengubah status pesanan menjadi sudah bayar serta mencetak setruk pembelian. Berikut *activity diagram* melihat dan mengubah status pesanan bagian kasir yang dapat dilihat pada Gambar 15.



Gambar 15. *Activity Diagram* Melihat dan Mengubah Status Pesanan Bagian Kasir.

### 11) *Activity Diagram* Mencetak Laporan Pembayaran

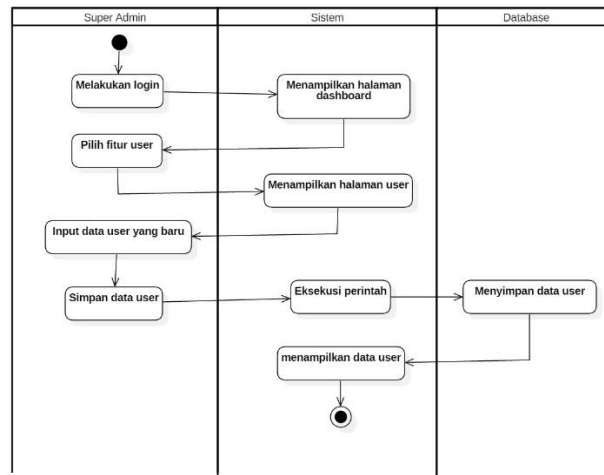
Super admin dan kasir dapat mencetak laporan sesuai dengan periode tanggal yang dipilih. Berikut *activity diagram* mencetak laporan pembayaran yang dapat dilihat pada Gambar 16.



Gambar 16. *Activity Diagram* Mencetak Laporan Pembayaran.

### 12) *Activity Diagram* Tambah User

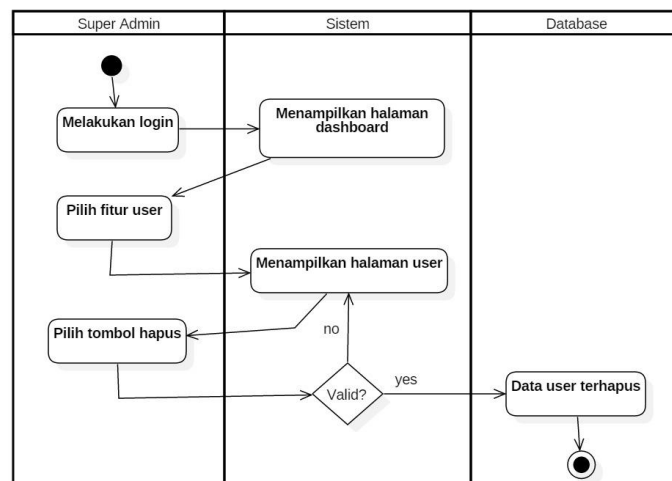
Jika ada karyawan baru dengan jabatan sebagai super admin, koki ataupun kasir akan ditambahkan akses masuk ke sistem oleh super admin yang lama untuk mendapatkan nama pengguna dan kata sandi. Berikut *activity diagram* tambah user yang dapat dilihat pada Gambar 17.



Gambar 17. *Activity Diagram Tambah User.*

### 13) *Activity Diagram Hapus User*

Selain dapat menambahkan *user*, super admin juga dapat menghapus *user* untuk melepaskan akses *user* lama dengan sistem. Berikut *activity diagram* hapus *user* yang dapat dilihat pada Gambar 18.



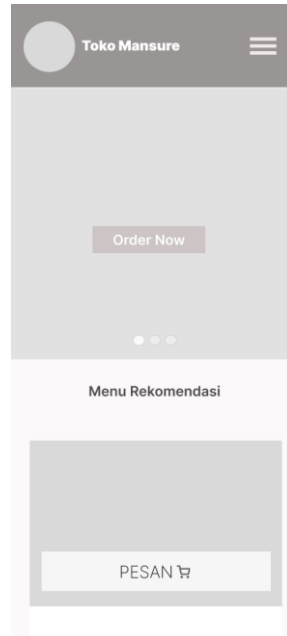
Gambar 18. *Activity Diagram Hapus User.*

### C. Desain antarmuka atau *User Interface*

Rancangan *user interface* menggambarkan tampilan dari sistem yang akan dibuat. Rancangan *user interface* pada penelitian ini adalah sebagai berikut.

### 1) Rancangan Tampilan Beranda

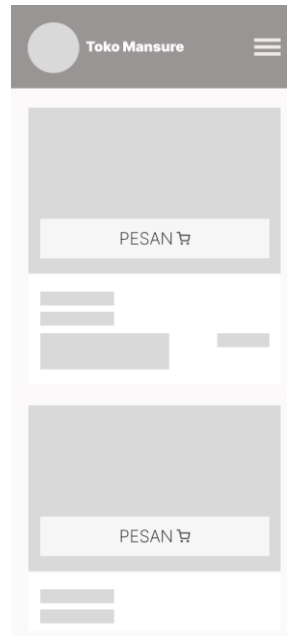
Saat pertama *user* mengakses *website* akan menampilkan halaman beranda, dimana tampilan beranda berisi daftar makanan dan minuman yang direkomendasikan oleh Toko Mansure. Berikut adalah rancangan tampilan beranda pelanggan yang ada pada Gambar 19.



Gambar 19. Rancangan Tampilan Beranda.

### 2) Rancangan Tampilan Menu

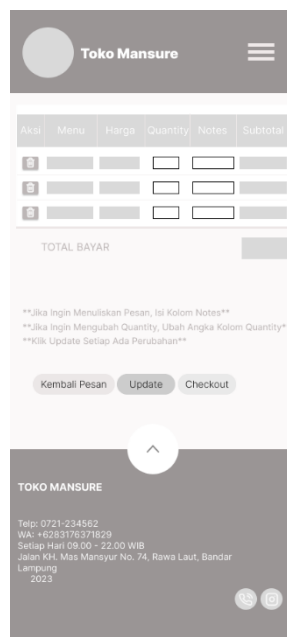
Pada halaman ini berisi informasi-informasi mengenai menu makanan dan minuman pada Toko Mansure. Berikut adalah rancangan tampilan menu yang ada pada Gambar 20.



Gambar 20. Rancangan Tampilan Menu.

### 3) Rancangan Tampilan Keranjang Pesanan

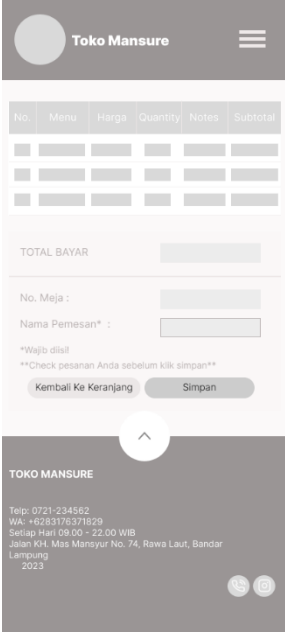
Modal ini muncul ketika pelanggan sudah klik pilih pada pesanan yang dipilih. Aksi yang bisa dilakukan oleh pelanggan adalah menghapus, mengedit pesanan. Berikut adalah rancangan tampilan keranjang pesanan yang ada pada Gambar 21.



Gambar 21. Rancangan Tampilan Keranjang Pesanan.

#### 4) Rancangan Tampilan Tagihan Pembayaran

Pada halaman ini menampilkan informasi-informasi mengenai pesanan yang sudah dipesan oleh pelanggan dan terdapat total bayar yang harus dibayarkan oleh pelanggan di kasir. Berikut adalah rancangan tampilan tagihan pembayaran pelanggan yang ada pada Gambar 22.

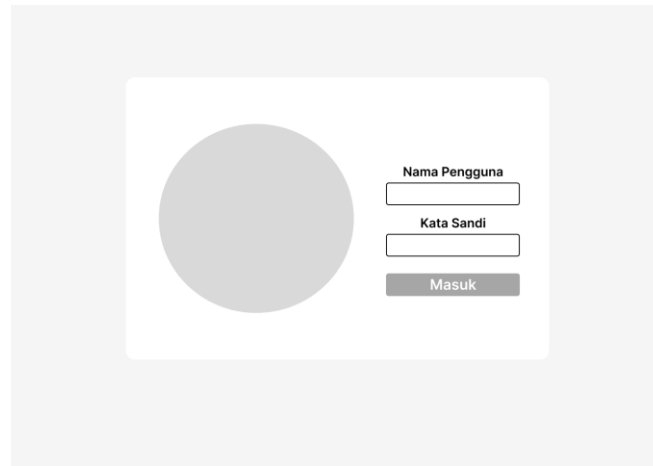


The image shows a mobile application interface for 'Toko Mansure'. At the top, there is a header with the store name and a menu icon. Below the header is a table with columns for 'No.', 'Menu', 'Harga', 'Quantity', 'Notes', and 'Subtotal'. The table contains several rows of placeholder data. Below the table, there is a section for 'TOTAL BAYAR' with a corresponding input field. Underneath, there are fields for 'No. Meja' and 'Nama Pemesan\*'. A note below these fields states '\*Wajib diisi' and '\*\*Check pesanan Anda sebelum klik simpan\*\*'. At the bottom of this section are two buttons: 'Kembali Ke Keranjang' and 'Simpan'. A circular arrow icon is positioned below the buttons. The footer of the application displays the store's name 'TOKO MANSURE', contact information (phone and WhatsApp), operating hours, and address in Lampung, 2023.

Gambar 22. Rancangan Tampilan Tagihan Pembayaran.

#### 5) Rancangan Tampilan *Login* Karyawan

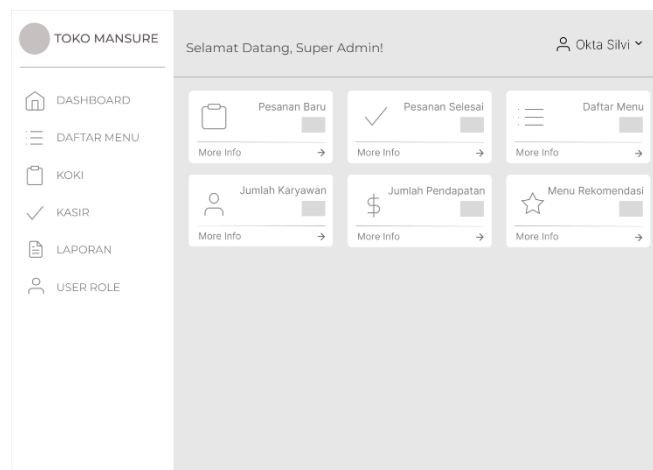
Halaman ini merupakan halaman yang diakses oleh koki, kasir dan super admin untuk masuk ke *dashboard*. Berikut adalah rancangan tampilan *login* karyawan yang ada pada Gambar 23.



Gambar 23. Rancangan Tampilan *Login* Karyawan.

#### 6) Rancangan Tampilan *Dashboard* Super Admin

Halaman ini adalah tampilan awal setelah super admin melakukan proses *login*. Semua fitur yang ada di *dashboard* dapat diakses oleh super admin. Berikut adalah rancangan tampilan *dashboard* Super Admin yang ada pada Gambar 24.



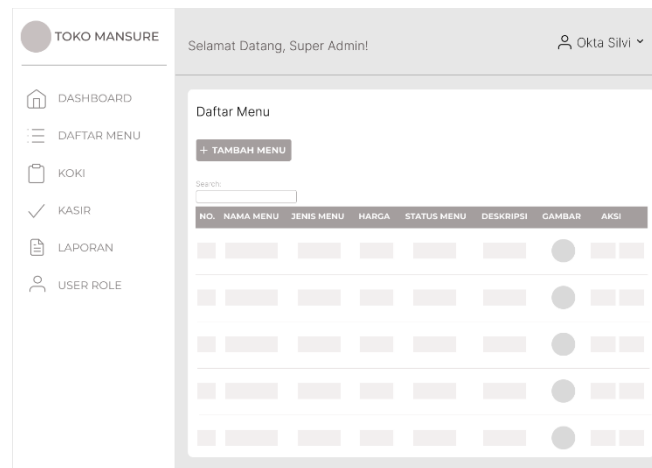
Gambar 24. Rancangan Tampilan *Dashboard* Super Admin.

#### 7) Rancangan Tampilan Daftar Menu

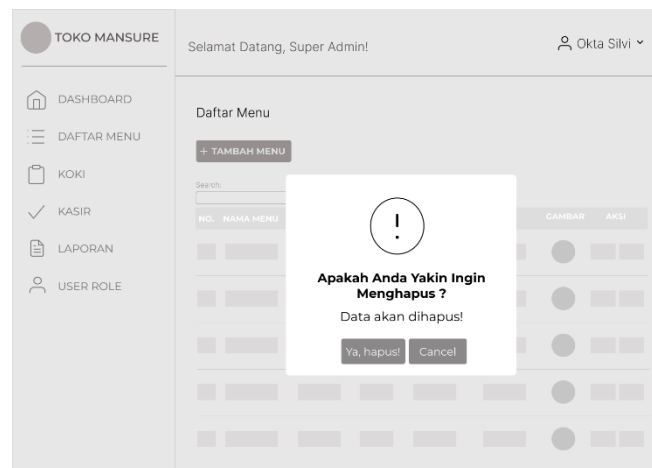
Halaman ini berisi *list* informasi daftar makanan dan minuman yang sudah diinputkan. Aksi yang dapat dilakukan adalah menambahkan, mengedit dan menghapus daftar menu. Halaman ini hanya bisa dikelola oleh super admin. Berikut



adalah rancangan tampilan daftar menu yang ada pada Gambar 25 dan rancangan tampilan hapus menu yang ada pada Gambar 26.



Gambar 25. Rancangan Tampilan Daftar Menu.



Gambar 26. Rancangan Tampilan Hapus Menu.

#### 8) Rancangan Tampilan Tambah Menu

Halaman ini untuk menambahkan menu-menu yang belum diinputkan. Super admin perlu memasukkan beberapa informasi mengenai menu yang akan ditambah. Berikut adalah rancangan tampilan tambah menu yang ada pada Gambar 27.

The screenshot shows the 'Tambah Menu' form in the TOKO MANSURE system. The form is titled 'Tambah Menu' and is located in the 'DAFTAR MENU' section of the dashboard. The form includes the following fields and controls:

- Nama Menu \***: A text input field with a placeholder 'Silahkan Masukkan Nama Menu'.
- Harga Menu \***: A text input field with a placeholder 'Silahkan Masukkan Harga Menu'.
- Deskripsi Menu**: A text input field with a placeholder 'Silahkan Masukkan Deskripsi Menu'.
- Jenis Menu \***: A dropdown menu with a placeholder 'Pilih'.
- Status Menu \***: A dropdown menu with a placeholder 'Pilih'.
- Jenis Beverages (isi Jika Jenis Menu Beverages)**: A dropdown menu with a placeholder 'Pilih'.
- Rekomendasi \***: A dropdown menu with a placeholder 'Pilih'.
- Upload Gambar \***: A file upload section with a 'Choose File' button and the text 'No File Chosen'.

At the bottom right of the form, there are two buttons: 'KEMBALI' and 'SIMPAN'. A note at the bottom left of the form states '\*Wajib Diisi'.

Gambar 27. Rancangan Tampilan Tambah Menu.

#### 9) Rancangan Tampilan Edit Menu

Halaman ini untuk mengedit menu-menu yang sudah diinputkan. Super admin perlu mengubah beberapa informasi mengenai menu yang akan diubah. Berikut adalah rancangan tampilan edit menu yang ada pada Gambar 28.

The screenshot shows the 'Edit Menu' form in the TOKO MANSURE system. The form is titled 'Edit Menu' and is located in the 'DAFTAR MENU' section of the dashboard. The form includes the following fields and controls:

- Nama Menu \***: A text input field.
- Harga Menu \***: A text input field.
- Deskripsi Menu**: A text input field.
- Jenis Menu \***: A dropdown menu.
- Status Menu \***: A dropdown menu.
- Jenis Beverages (isi Jika Jenis Menu Beverages)**: A dropdown menu.
- Rekomendasi \***: A dropdown menu.
- Upload Gambar \***: A file upload section with a 'Choose File' button and the text 'No File Chosen'.

At the bottom right of the form, there are two buttons: 'KEMBALI' and 'SIMPAN'.

Gambar 28. Rancangan Tampilan Edit Menu.

#### 10) Rancangan Tampilan *User Role*

Halaman ini digunakan untuk menambahkan karyawan baru Toko Mansure sesuai dengan jabatannya dan juga terdapat list daftar karyawan yang sudah terdaftar. Halaman ini hanya bisa dikelola oleh super admin. Pada halaman ini terdapat 2 aksi yaitu menghapus akun *user* dan mereset *password user*. Berikut

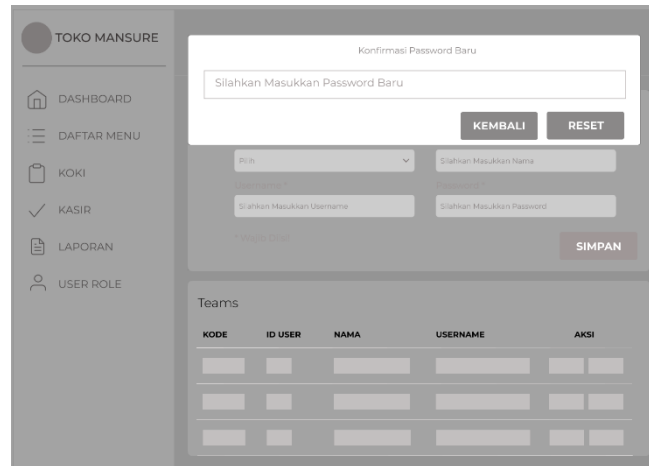
adalah rancangan tampilan *user* yang ada pada Gambar 29, rancangan tampilan hapus akun *user* yang ada pada Gambar 30 dan rancangan tampilan reset *password user* yang ada pada Gambar 31.

The screenshot shows the 'Tambah User' form in the TOKO MANSURE dashboard. The dashboard header includes the logo 'TOKO MANSURE', the greeting 'Selamat Datang, Super Admin!', and the user profile 'Okta Silvi'. The left sidebar contains navigation items: DASHBOARD, DAFTAR MENU, KOKI, KASIR, LAPORAN, and USER ROLE. The main content area features a 'Tambah User' form with the following fields: 'Kode \*' (a dropdown menu), 'Nama \*' (text input), 'Username \*' (text input), and 'Password \*' (text input). Below the form is a 'Teams' table with columns 'KODE', 'ID USER', 'NAMA', 'USERNAME', and 'AKSI'. The table contains three rows of placeholder data.

Gambar 29. Rancangan Tampilan *User Role*.

The screenshot shows the 'Hapus Akun' confirmation dialog in the TOKO MANSURE dashboard. The dialog has a title bar 'Apakah Anda Yakin Akan Menghapus Akun Okta Silvi?' and two buttons: 'KEMBALI' and 'HAPUS'. Below the dialog is the 'Tambah User' form, which is dimmed. The form fields are: 'Kode \*' (dropdown), 'Nama \*' (text input), 'Username \*' (text input), and 'Password \*' (text input). Below the form is the 'Teams' table with columns 'KODE', 'ID USER', 'NAMA', 'USERNAME', and 'AKSI'. The table contains three rows of placeholder data.

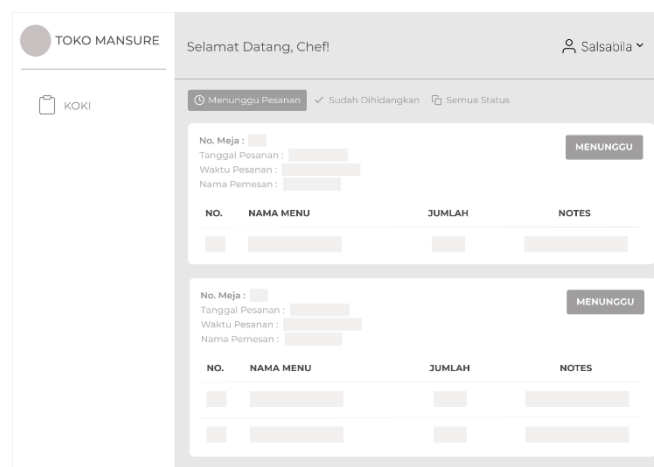
Gambar 30. Rancangan Tampilan Hapus Akun *User*.



Gambar 31. Rancangan Tampilan Reset *Password User*.

### 11) Rancangan Tampilan Koki

Halaman ini berisi data pesanan yang masuk dari pelanggan. Aksi yang bisa dilakukan oleh koki adalah mengklik status pesanan “menunggu” menjadi “selesai” jika pesanan sudah siap dihidangkan, kemudian data pesanan akan diteruskan ke bagian kasir. Halaman ini hanya bisa dikelola oleh super admin dan koki. Berikut adalah rancangan tampilan koki yang ada pada Gambar 32.

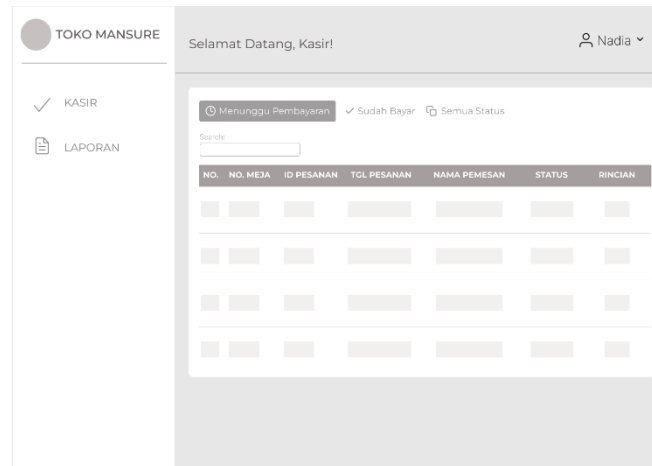


Gambar 32. Rancangan Tampilan Koki.

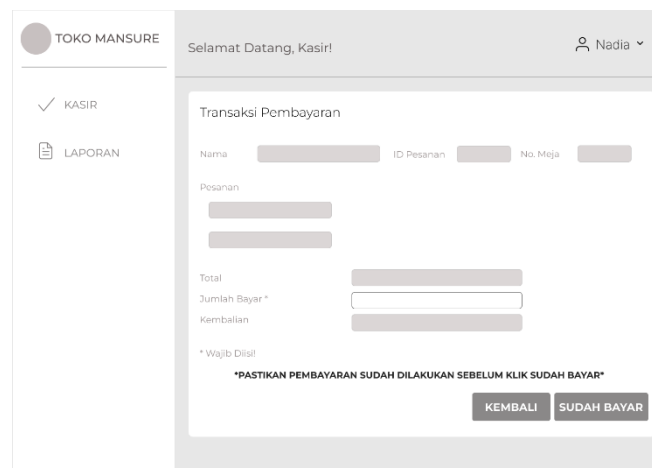
### 12) Rancangan Tampilan Transaksi

Halaman ini berisi data pesanan yang sudah dihidangkan dari bagian koki. Aksi yang bisa dilakukan oleh kasir adalah menampilkan transaksi pembayaran untuk

menginputkan jumlah yang dibayarkan pelanggan dan status pesanan otomatis berubah menjadi “sudah bayar”. Halaman ini hanya bisa dikelola oleh super admin dan kasir. Berikut adalah rancangan tampilan kasir yang ada pada Gambar 33 dan rancangan tampilan transaksi pembayaran yang ada pada Gambar 34.



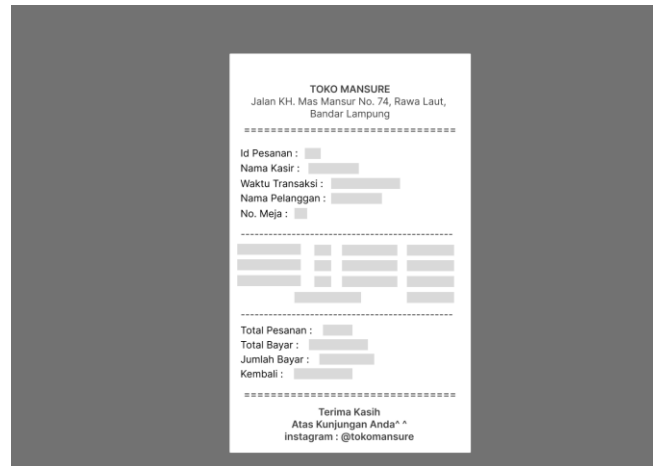
Gambar 33. Rancangan Tampilan Kasir.



Gambar 34. Rancangan Tampilan Transaksi Pembayaran.

### 13) Rancangan Tampilan Setruk Pembelian

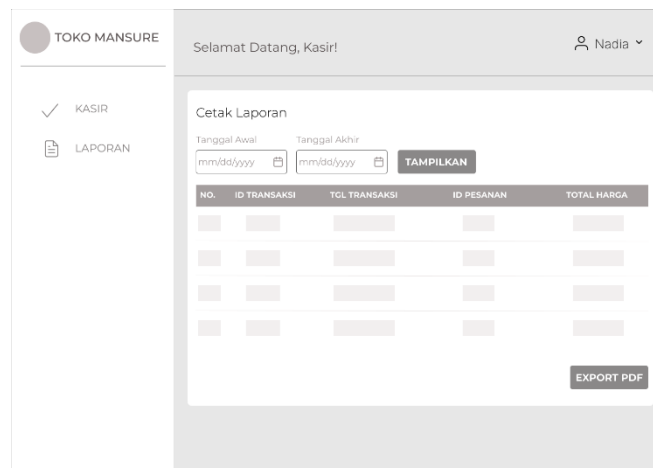
Halaman ini menampilkan informasi pesanan serta total pembayaran dan dapat dicetak untuk diberikan kepada pelanggan yang sudah melakukan pembayaran. Berikut adalah rancangan tampilan setruk pembelian yang ada pada Gambar 35.



Gambar 35. Rancangan Tampilan Setruk Pembelian.

#### 14) Rancangan Tampilan Cetak Laporan

Halaman ini digunakan untuk melihat dan mencetak laporan transaksi pembayaran sesuai dengan periode tanggal yang diinputkan. Halaman ini hanya bisa dikelola oleh super admin dan kasir. Berikut adalah rancangan tampilan cetak laporan yang ada pada Gambar 36 dan rancangan tampilan hasil *export* laporan yang ada pada Gambar 37.



Gambar 36. Rancangan Tampilan Cetak Laporan.



## 2. Pengujian Kemudahan Pengguna

Pada tahapan ini dilakukan menguji sistem yang sudah dibuat menggunakan metode *usability testing*. Pada pengujian ini ada beberapa tahapan, yaitu dengan menentukan skenario pengujian, memilih responden, pengujian oleh responden dan rekapitulasi hasil (Fatmawati, 2021).

### a. Menentukan Skenario Pengujian

Skenario pengujian menjelaskan tentang sistem yang akan diuji dan *user* harus melakukan percobaan sistem sebelum melakukan pengisian kuesioner. Berikut skenario pengujian pada modul pelanggan, kasir, koki dan super admin dijelaskan pada Tabel 3, Tabel 4, Tabel 5, Tabel 6 secara berturut.

Tabel 3. Skenario Pengujian Modul Pelanggan

No.	Kelas Uji	Daftar Pengujian	Kasus Uji
1.	Menu rekomendasi	1. Melihat menu yang direkomendasikan.	a. Mengakses halaman beranda.
2.	Melakukan pemesanan	1. Melihat menu makanan dan minuman. 2. Memilih pesanan makanan dan minuman. 3. Melihat pesanan yang sudah dipilih. 4. Menghapus pesanan yang sudah dipilih. 5. Mengedit jumlah pesanan yang sudah dipilih 6. Menambahkan catatan pesanan 7. Menambahkan pesanan kembali 8. Menyimpan pesanan	a. Mengakses halaman menu. b. Memilih sesuai jenis menu pada dropdown menu. c. Menekan tombol pilih pada menu yang diinginkan. d. Mengakses halaman keranjang e. Menekan <i>icon</i> hapus pada kolom aksi. f. Mengubah <i>field quantity</i> . g. Menekan tombol <i>update</i> . i. Mengisi <i>field note</i> dengan benar. j. Menekan tombol <i>update</i> . k. Menekan tombol kembali pesan dan memilih pesanan kembali. l. Menekan tombol <i>checkout</i> .
3.	Tagihan Pembayaran	1. Melihat tagihan pembayaran. 2. Menginputkan nama pelanggan 3. Kembali ke halaman keranjang 4. Menyimpan pesanan	a. Mengakses halaman tagihan. b. Mengisi <i>field</i> nama pelanggan. c. Menekan tombol kembali ke keranjang. d. Menekan tombol simpan.



Tabel 4. Skenario Pengujian Modul Koki

No.	Kelas Uji	Daftar Pengujian	Kasus Uji
1.	<i>login</i>	1. Melakukan <i>login</i> .	a. Mengisi <i>field username</i> dan <i>password</i> dengan benar dan lengkap. b. Mengisi <i>field username</i> dan <i>password</i> dengan salah dan tidak lengkap. c. Menekan tombol <i>login</i> .
2.	Mengelola pesanan	1. Melihat pesanan yang masuk. 2. Menyelesaikan pesanan yang sudah dibuat.	a. Mengakses halaman koki. b. Menekan tombol menunggu pada masing-masing pesanan. c. Status berubah menjadi "Selesai".
3.	<i>logout</i>	1. Melakukan <i>logout</i> .	a. Menekan nama akun. b. Menekan tombol <i>logout</i> .

Tabel 5. Skenario Pengujian Modul Kasir

No.	Kelas Uji	Daftar Pengujian	Kasus Uji
1.	<i>login</i>	1. Melakukan <i>login</i> .	a. Mengisi <i>field username</i> dan <i>password</i> dengan benar dan lengkap. b. Mengisi <i>field username</i> dan <i>password</i> dengan salah dan tidak lengkap. c. Menekan tombol <i>login</i> .
2.	Mengelola transaksi	1. Melihat pesanan yang sudah diproses. 2. Melakukan pembayaran. 3. Mencetak setruk pembayaran.	a. Mengakses halaman kasir. b. Menekan icon pensil pada rincian. c. Menginputkan jumlah bayar pada <i>field</i> jumlah bayar. d. Menekan tombol sudah bayar. e. Setelah klik sudah bayar kemudian menekan tombol print.
3.	Mengelola laporan transaksi	1. Melihat laporan transaksi sesuai dengan periode yang diinputkan. 2. Mencetak laporan transaksi.	a. Mengakses halaman laporan. b. Memilih tanggal periode dari dan sampai tanggal laporan. c. Menekan tombol tampilkan. d. Menekan tombol <i>export</i> PDF.
4.	<i>logout</i>	1. Melakukan <i>logout</i> .	a. Menekan nama akun. b. Menekan tombol <i>logout</i> .

Tabel 6. Skenario Pengujian Modul Super Admin

No.	Kelas Uji	Daftar Pengujian	Kasus Uji
1.	<i>login</i>	1. Melakukan login.	a. Mengisi <i>field username</i> dan <i>password</i> dengan benar dan lengkap. b. Mengisi <i>field username</i> dan <i>password</i> dengan salah dan tidak lengkap. c. Menekan tombol <i>login</i> .
2.	Mengelola daftar menu	1. Melihat seluruh daftar menu. 2. Menambahkan menu.  3. Mengedit menu.  4. Menghapus menu.  5. Mencari data.	a. Mengakses halaman daftar menu.  b. Menekan tombol tambah menu. c. Mengisi <i>field</i> dengan lengkap dan benar. d. Mengisi <i>field</i> dengan tidak lengkap dan salah. e. Menekan tombol simpan. f. Menekan <i>icon</i> pensil pada kolom rincian. g. Mengubah <i>field</i> . h. Tidak mengisi salah satu <i>field</i> . i. Menekan tombol simpan. j. Menekan <i>icon</i> hapus pada kolom rincian.  i. Menekan kotak <i>search</i> . j. Menginputkan data yang dicari berdasarkan nama, jenis, status dan deskripsi menu.
3.	Mengelola pesanan	1. Melihat pesanan yang masuk. 2. Menyelesaikan pesanan yang sudah dibuat.	a. mengakses halaman koki. b. Menekan tombol menunggu pada masing-masing pesanan. c. Status berubah menjadi "Selesai".
4.	Mengelola transaksi	1. Melihat pesanan yang sudah diproses. 2. Melakukan pembayaran.  3. Mencetak setruk pembayaran.	a. Mengakses halaman kasir.  b. Menekan <i>icon</i> pensil pada rincian. c. Menginputkan jumlah bayar pada <i>field</i> jumlah bayar. d. Menekan tombol sudah bayar. e. Setelah klik sudah bayar kemudian menekan tombol print.
5.	Mengelola laporan transaksi	1. Melihat laporan transaksi sesuai dengan periode yang diinputkan.  2. Mencetak laporan transaksi.	a. Mengakses halaman laporan. b. Memilih tanggal periode dari dan sampai tanggal laporan. c. Menekan tombol tampilkan. d. Menekan tombol <i>export</i> PDF.
6.	Mengelola akun <i>user</i>	1. Menambahkan <i>user</i> .	a. Mengakses halaman <i>user role</i> . b. Mengisi <i>field</i> dengan lengkap dan benar. c. Mengisi <i>field</i> dengan tidak lengkap dan salah. d. Menekan tombol simpan.

No.	Kelas Uji	Daftar Pengujian	Kasus Uji
		2. Mereset <i>password user</i> .	e. Melihat tabel Teams. f. Menekan <i>icon</i> reset pada kolom aksi. g. Memasukkan <i>password</i> baru pada modal konfirmasi <i>password</i> . h. Menekan tombol simpan.
		3. Menghapus akun <i>user</i> .	i. Menekan <i>icon</i> hapus pada kolom aksi.
7.	logout	1. Melakukan <i>logout</i> .	a. Menekan nama akun. b. Menekan tombol <i>logout</i> .

#### b. Memilih Responden

Pada tahapan ini dilakukan menentukan responden yang akan dilibatkan dalam pengujian sistem pemesanan makanan dan minuman pada Toko Mansure, yaitu pelanggan dan karyawan Toko Mansure.

#### c. Pengujian oleh Responden

Pada tahapan ini, responden melaksanakan skenario pengujian yang telah diberikan kemudian memberikan penilaian terhadap sistem melalui instrumen SUS. Berikut adalah instrumen SUS yang dapat dilihat pada Tabel 7.

Tabel 7. Instrumen SUS

No.	Pernyataan	Skala
1.	Saya pikir bahwa saya akan menginginkan lebih sering menggunakan sistem ini	1 – 5
2.	Saya menemukan bahwa sistem ini, tidak harus dibuat serumit ini	1 – 5
3.	Saya pikir sistem mudah untuk digunakan	1 – 5
4.	Saya pikir bahwa saya akan membutuhkan bantuan dari orang teknis untuk dapat menggunakan sistem ini	1 – 5
5.	Saya menemukan berbagai fungsi di sistem ini diintegrasikan dengan baik	1 – 5
6.	Saya pikir ada terlalu banyak ketidakkonsistenan dalam sistem ini	1 – 5
7.	Saya bayangkan bahwa kebanyakan orang akan mudah untuk mempelajari sistem ini dengan sangat cepat	1 - 5
8.	Saya menemukan, sistem ini sangat rumit untuk digunakan	1 – 5
9.	Saya merasa sangat percaya diri untuk menggunakan sistem ini	1 – 5
10.	Saya perlu belajar banyak hal sebelum saya bisa memulai menggunakan sistem	1 – 5

Skala jawaban pada kuesioner SUS meliputi angka 1 sampai 5 dengan kriteria penilaian yang dapat dilihat pada Tabel 8.

Tabel 8. Kriteria Penilaian

Nilai	Kriteria
1	Sangat Tidak Setuju
2	Tidak Setuju
3	Ragu-ragu
4	Setuju
5	Sangat Setuju

#### d. Rekapitulasi Hasil

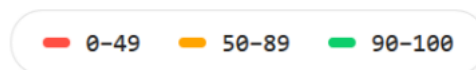
Setelah menentukan skenario pengujian, memilih responden dan melakukan pengujian, selanjutnya adalah merekap hasil pengujian menggunakan perhitungan SUS. Adapun ketentuan dalam perhitungan hasil evaluasi dengan SUS yaitu sebagai berikut (Fatmawati, 2021).

- 1) Skala jawaban dikurangi 1 pada pernyataan ganjil
- 2) 5 dikurangi skala jawaban pada pernyataan genap
- 3) Skala jawaban dijumlahkan dengan dikali dengan angka 2,5
- 4) Rata-rata jawaban ditentukan dari pernyataan.

Ada 3 aspek utama tingkat penilaian, yang pertama adalah *acceptability* (penerimaan) yaitu menentukan apakah sistem informasi dapat diterima oleh pengguna, dengan tingkatan yang meliputi *not acceptable*, *marginal (low dan high)* dan *acceptable*. Kedua, *grade scale* (skala kelas) yaitu menentukan bagaimana kualitas sistem informasi, dengan tingkatan yang meliputi A, B, C, D dan F. Ketiga, *adjective rating* (peringkat) yaitu menentukan peringkat sistem informasi berguna atau tidak, dengan tingkatan yang meliputi *worst imaginable*, *poor*, *ok*, *good*, *excellent* dan *best imaginable*. Rekapitulasi nilai berupa nilai kumulatif rata-rata instrumen akan menentukan hasil evaluasi sistem. Hasil evaluasi berupa *acceptability*, *grade scale* dan *adjective rating*, bukan dari nilai masing-masing instrumen SUS (Fatmawati, 2021). Penentuan hasil evaluasi *usability* dengan SUS dapat dilihat pada Gambar 2.

### 3. Pengujian Kecepatan

Pengujian kecepatan dilakukan menggunakan *tools lighthouse*. Halaman yang akan diaudit dijalankan dengan lighthouse, kemudian lighthouse menghasilkan laporan yang menunjukkan seberapa baik performa halaman tersebut. Audit yang gagal dapat digunakan sebagai indikator untuk meningkatkan halaman (Google, 2016).



Gambar 38. *Parameter Score*.

Pada Gambar 38 menunjukkan index parameter yang digunakan untuk menentukan kualitas halaman yang diukur. Skor parameter 0-49 menunjukkan bahwa halaman tersebut memiliki performa yang buruk dan membutuhkan banyak yang harus dioptimalkan. Skor 50-59 menunjukkan bahwa halaman tersebut berada ukuran normal, tetapi masih ada yang harus dioptimalkan. Skor 90-100 menunjukkan bahwa halaman tersebut memiliki performa yang sangat baik (Anwar et al., 2020). Pengukuran performansi website menggunakan 5 kriteria yang ditunjukkan pada Tabel 9 (Wisnuadhi et al., 2022).

Tabel 9. Evaluasi Parameter *Performance Metric*.

Kriteria	Deskripsi	Indikator	Tingkat Performansi
<i>First Contentful Paint</i> (FCP)	Ukuran waktu <i>website</i> menampilkan konten pertama yang berupa gambar dan teks.	0 - 1.8 detik	cepat, indikator berwarna hijau
		1.8 - 3 detik	sedang, indikator berwarna jingga.
		> 3 detik	lambat, indikator berwarna merah.
<i>Speed Index</i> (SI)	Ukuran kecepatan waktu <i>website</i> menampilkan konten secara visual selama <i>page load</i> .	0 - 3.4 detik	cepat, indikator berwarna hijau.
		3.4 - 5.8 detik	sedang, indikator berwarna jingga.
		> 5.8 detik	lambat, indikator berwarna merah.
<i>Largest Contentful Paint</i> (LCP)	Ukuran waktu <i>website</i> menampilkan konten terbesar pada halaman <i>viewpoint</i> saat <i>rendering</i> .	0 - 2.5 detik	cepat, indikator berwarna hijau.
		2.5 - 4 detik	sedang, indikator berwarna jingga.
		> 4 detik	lambat, indikator berwarna merah,

<i>Total Blocking Time (TBT)</i>	Total waktu <i>website</i> memblokir semua respon input pengguna saat halaman sedang dimuat sampai fungsi interaktif berjalan. Dengan kata lain, jumlah total waktu yang dibutuhkan antara FCP dan TTI.	0 - 200 milidetik 200 - 600 milidetik > 600 milidetik	cepat, indikator berwarna hijau. sedang, indikator berwarna jingga. lambat, indikator berwarna merah.
<i>Cumulative Layout Shift (CLS)</i>	Ukuran skor kumulatif elemen <i>website</i> pergeseran tak terduga di halaman <i>viewpoint</i> . Pengukuran <i>page load</i> sampai semua status <i>lifecycle</i> tersembunyi.	0 - 0.1 0.1 -2.5 > 2.5	bagus, indikator berwarna hijau. perlu perbaikan, indikator berwarna jingga. buruk, indikator berwarna merah.

#### 4. Pengujian Aksesibilitas

Pengujian aksesibilitas dilakukan menggunakan *tools* aXe DevTools dari Deque dengan masuk ke panel DevTools untuk menemukan masalah aksesibilitas pada sistem yang dibuat.

#### 3.2.3. Tahapan Penulisan Laporan

Pada tahapan ini merupakan dokumentasi serta menjelaskan bagaimana membangun sistem informasi berbasis *website* sesuai dengan alur tahapan penelitian. Berikut adalah alur tahapan penelitian yang ada pada Tabel 10.



## V. KESIMPULAN DAN SARAN

### 5.1. Kesimpulan

Berdasarkan dari hasil dan pembahasan diatas, maka dapat disimpulkan bahwa penelitian ini adalah sebagai berikut.

1. Telah terbangunnya *front-end* pada Digitalisasi Pemesanan Menu Makanan dan Minuman berbasis *Website* pada Toko Mansure.
2. *Front-end* pada sistem ini telah dilakukan pengujian menggunakan metode *usability testing* dengan kuesioner *System Usability Scale (SUS)* yang mendapatkan nilai akhir 80, dengan kriteria *adjective rating* (peringkat) *good*, *grade scale* (skala kelas) B dan *acceptability ranges* (penerimaan) *acceptable*.
3. *Front-end* sistem ini telah dilakukan pengujian aksesibilitas menggunakan aXe DevTools dan mendapatkan 1 (satu) *issue* yaitu *id attribute value must be unique* pada halaman keranjang saya, daftar menu dan koki.
4. Sistem informasi ini dapat membantu pelanggan dalam melakukan pemesanan makanan dan minuman.
5. Pembangunan *front-end* sistem informasi ini memudahkan pengembangan *back-end* untuk dapat difungsikan.

### 5.2. Saran

Berdasarkan hasil implementasi sistem yang telah dibuat, maka dalam penelitian ini disarankan untuk sebagai berikut.



1. Menggunakan *two factor authentication* atau dua jenis autentikasi pada saat melakukan reset *password user* koki, kasir maupun super admin sehingga dapat memperkuat keamanan akun *user*.
2. Disarankan untuk menambahkan fitur estimasi waktu pesanan datang untuk pelanggan dapat memprediksi pesanan datang juga dapat meningkatkan kepuasan pelanggan.
3. Sistem ini dimungkinkan untuk dikembangkan selanjutnya untuk dapat melakukan pemesanan dari rumah dan melakukan *food delivery*.

## DAFTAR PUSTAKA

- Anggraeni, E. Y., & Irviani, R. (2017). *PENGANTAR SISTEM INFORMASI* (E. Risanto (Ed.); 1st ed.). Penerbit Andi.
- Anwar, S., Permana, A. G., & Darlis, D. (2020). RANCANG BANGUN DAN IMPLEMENTASI SISTEM INFORMASI PENGELOLAAN SAMPAH DAN TEMPAT OLAH SAMPAH SETEMPAT TERPADU MANDIRI (TOSS TM) DENGAN KONTROL TERPUSAT BERBASIS INTERNET OF THINGS. *E-Proceeding of Applied Science*, 6(1), 660–673.
- Aprilia, P. (2021). *Apa itu HTML? Berikut Fungsi dan Cara Kerjanya!* Niagahoster. <https://www.niagahoster.co.id/blog/html-adalah/>
- Asaniyah, N. (2017). *PELESTARIAN INFORMASI KOLEKSI LANGKA : Digitalisasi, Restorasi, Fumigasi*. 85–94.
- Asyikin, A. N. (2019). *PEMROGRAMAN WEB* (1st ed.). POLIBAN PRESS.
- Azzis, N. F. (2022). *SISTEM INFORMASI PENJUALAN MAKANAN BERBASIS WEB PADA RUMAH MAKAN JAGOSORE*. Universitas Muhammadiyah Surakarta.
- Bader, W. I., & Hammouri, A. I. (2016). Responsive Web Design Techniques. *International Journal of Computer Applications*, 150(2). <https://doi.org/10.5120/ijca2016911463>
- Deque. (n.d.). *axe DevTools Chrome browser extension*. Deque. Retrieved July 1, 2023, from <https://www.deque.com/axe/devtools/chrome-browser-extension/>
- Dequeuniversity. (n.d.). *Id Attribute Value Must be Unique*. Dequeuniversity.Com. <https://dequeuniversity.com/rules/axe/4.7/duplicate-id?application=AxeChrome>

- Dharma, R. (2021). *Front End Adalah: Pengertian, Cara Kerja, dan Perbedaannya dengan Back End*. Accurate.Id.  
<https://accurate.id/teknologi/front-end-adalah/>
- Ependi, U., Kurniawan, T. B., & Panjaitan, F. (2019). SYSTEM USABILITY SCALE VS HEURISTIC EVALUATION : A REVIEW. *Jurnal SIMETRIS*, 10(1), 65–74.
- Fatmawati, A. (2021). Evaluasi Usability pada Learning Management System OpenLearning Menggunakan System Usability Scale. *Jurnal Inovtek Polbeng - Seri Informatika*, 6(1).
- Fatta, H. Al. (2007). *Analisis dan Perancangan Sistem Informasi untuk Keunggulan Bersaing Perusahaan dan Organisasi Modern* (A. H. Triyuliana (Ed.); 1st ed.). Penerbit Andi.
- Fauzi, D., & Ismatullah, I. (2022). *Belajar HTML dan CSS Dasar : Studi Kasus* (P. V. Buana (Ed.); 1st ed.). Yayasan Pendidikan Cendekia Muslim.
- Google. (2016). *Overview*. Chrome Developers.  
<https://developer.chrome.com/docs/lighthouse/overview/>
- Gravityforms. (2022). *Accessibility Guide for Developers*. Gravity Forms.  
<https://docs.gravityforms.com/accessibility-for-developers/#h-axe-developer-tools>
- Hafiar, H., Subekti, P., Setianti, Y., & Amin, K. (2022). *Aksesibilitas website dan reputasi online marketplace reksadana*. 6(2), 197–219.
- Handayani, T., Gunawan, I., & Taufiq, R. (2020). RANCANG BANGUN SISTEM INFORMASI PEMESANAN MENU MAKANAN BERBASIS WEB (STUDI KASUS: RESTORAN BUKIT RANDU BANDARA). *Jurnal SITECH*, 3(1).
- Handiwidjojo, W., & Ernawati, L. (2016). Pengukuran Tingkat Ketergunaan (Usability) Sistem Informasi Keuangan Studi Kasus : Duta Wacana Internal Transaction (Duwit). *Jurnal Ilmiah Sistem Informasi*, 02(01).
- Henim, S. R., Muslim, I., & Sari, R. P. (2023). PENERAPAN WEBSITE ACCESSIBILITY CONFORMANCE EVALUATION METHODOLOGY (WCAG-EM) UNTUK MENGUKUR ACCESSIBILITY WEBSITE BP3M

POLITEKNIK CALTEX RIAU. *Journal of Information Technology and Computer Science (INTECOMS)*, 6(1).

Hristov, H., Stoitsov, G., Bliznakov, M., & Minchev, S. (2023). An Approach to Validating the Accessibility of Dynamic Web Pages. *TEM Journal*, 12(1).  
<https://doi.org/10.18421/TEM121>

Hutahaean, J. (2014). *Konsep Sistem Informasi* (1st ed.). Deepublish.

Iqbal, M. (2022). *Website Adalah: Pengertian Menurut Para Ahli, Fungsi, Jenis dan Manfaatnya (Lengkap 2022)*. Lindungihutan.  
<https://lindungihutan.com/blog/pengertian-website-adalah/#rb-pengertian-website-menurut-ahli>

Ismail, A., & Kuppusamy, K. S. (2019). Web accessibility investigation and identification of major issues of higher education websites with statistical measures : A case study of college websites. *Journal of King Saud University - Computer and Information Sciences*, 34(3), 901–911.  
<https://doi.org/10.1016/j.jksuci.2019.03.011>

Isnainrajab, I., Wijoyo, S. H., & Perdanakusuma, A. R. (2020). Evaluasi Usability Pada Aplikasi PermataMobile X Dengan Menggunakan Metode Usability Testing Dan System Usability Scale (SUS). *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, 4(10), 3383–3391. <http://j-ptiik.ub.ac.id>

Juviler, J. (2021). *How to Use Google Lighthouse to Optimize Your Website UX*. Hubspot. <https://blog.hubspot.com/website/google-lighthouse>

Kurniawan, T. B., & Syarifuddin. (2020). PERANCANGAN SISTEM APLIKASI PEMESANAN MAKANAN DAN MINUMAN PADA CAFETARIA NO CAFFE DI TANJUNG BALAI KARIMUN MENGGUNAKAN BAHASA PEMROGRAMAN PHP DAN MYSQL. *Jurnal Tikar*, 1(2), 192–206.

*Lighthouse*. (2016). <https://developer.chrome.com/docs/lighthouse/overview/>

Logix. (2021). *Responsively App Is A Browser For Fast Responsive Web Development (Free And Open Source)*. Linuxuprising.  
<https://www.linuxuprising.com/2021/11/responsively-app-is-browser-for-faster.html>

Mobile, J. (2023). *Mengenal 7 Tugas Frontend Developer*. Jatis Mobile.

<https://jatismobile.com/id/blog/aplikasi-dan-platform-digital/tugas-frontend-developer/>

- Morris, W. (2022). *What Is Google Lighthouse and How to Use It?* Elegantthemes. <https://www.elegantthemes.com/blog/wordpress/what-is-google-lighthouse-and-how-to-use-it>
- Novianty, C. (2017). REVIEW KONSEP RESPONSIVE DESIGN DENGAN FRAMEWORK MATERIALIZE PADA WEBSITE. *Jurnal Nasional Informatika Dan Teknologi Jaringan*, 2(1), 41–44.
- Nugroho, A. (2010). *Rekayasa Perangkat Lunak Berorientasi Objek dengan Metode USDP* (O. HS (Ed.); 1). Penerbit Andi.
- Patria, R. (2021). *12 Tugas dan Skill Front End Developer yang Wajib Kamu Tahu*. Domainsia.Com. <https://www.domainsia.com/berita/front-end-developer/>
- Pol, T. (2023). *Google Lighthouse: What It Is & How to Use It*. Semrushblog.
- Pribadi, J. D., Rohman, F., Suwarni, E., Pradana, K. R., & Lestanti, C. Q. R. O. (2022). *Sistem Informasi Agenda* (K. Ummatin (Ed.)). Jakad Media Publishing.
- Rizaldy, R., & Dirgahayu, R. T. (2020). Pengembangan Front-End Sistem Informasi Pendataan Pendar Foundation Yogyakarta. *AUTOMATA*, 1(2).
- Sari, I. P. (2021). *Buku Ajar Rekayasa Perangkat Lunak* (F. V. Riza (Ed.)). UMSU PRESS.
- Sauro, J. (2011). *Measuring Usability with the System Usability Scale (SUS)*. MeasuringU. <https://measuringu.com/sus/>
- Setiyani, H. (2019). *APLIKASI PEMESANAN MAKANAN BERBASIS WEB DI RAMEN SHINJU*. 5(3), 1982.
- Sidik, A. (2018). Penggunaan System Usability Scale ( SUS ) Sebagai Evaluasi Website Berita Mobile. *Jurnal Ilmiah "Technologia,"* 9(2), 83–88.

- Suryantara, I. G. N. (2017). *Merancang Aplikasi dengan Metodologi Extreme Programmings*. PT Elex Media Komputindo.
- Syakra, Y., Mariami, I., Murniyanti, S., Mahyuni, R., Yakub, S., Purba, M. H. G., & Siregar, M. A. (2022). Digitalisasi Infaq dan Shadaqah di Masjid Al-Chairat Berbasis Web. *ABDI SABHA (Jurnal Pengabdian Kepada Masyarakat)*, 3(2), 285–292.
- W3schools. (n.d.). *CSS Media Queries - Examples*. W3schools.Com. [https://www.w3schools.com/css/css3\\_mediaqueries\\_ex.asp](https://www.w3schools.com/css/css3_mediaqueries_ex.asp)
- Wahyudi, M., Hati, K., Larasati, F. B., Ismail, J., & Solikhun. (2021). *Fullstack Android Developer Aplikasi Penjualan Tiket Bioskop* (R. Watrionthos & J. Simarmata (Eds.); 1st ed.). Yayasan Kita Menulis.
- Wasino, Herwindiati, D. E., Setyawan, I. R., & Maupa, H. (2023). Desain Situs Web Yang Responsif Berdasarkan Strategi Agile Sebagai Pendukung Pemasaran Destinasi Wisata. *Jurnal Teknik Informatika Dan Sistem Informasi*, 10(1), 526–540.
- Wisnuadhi, B., Maspupah, A., Wulan, S. R., Sholahuddin, M. R., Setiarini, S. D., & Firdaus, L. H. (2022). Studi Komparatif Fungsionalitas , Performance dan User experience pada Website Geopark. *Journal of Information System Research (JOSH)*, 3(4), 579–589. <https://doi.org/10.47065/josh.v3i4.1879>
- Yasin. (2019). *10 Skill Yang Harus Dimiliki Front End Developer*. Niagahoster.Co.Id. <https://www.niagahoster.co.id/blog/skill-front-end-developer/>
- Yuhefizar, Mooduto, H., & Hidayat, R. (2009). *Cara Mudah Membangun website Interaktif Menggunakan Content Management System Joomla (CMS)* (Edisi Revi). PT Elex Media Komputindo.