

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

(Skripsi)

Oleh

**NADIA SALSABILA
NPM 1917051046**



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

ABSTRAK

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

Oleh

NADIA SALSABILA

Toko Mansure adalah salah satu kafe yang ada di Kota Bandar Lampung dengan pengunjung yang ramai. Memiliki lokasi yang strategis dan bangunan dengan konsep alam adalah salah satu penyebab Toko Mansure ramai pengunjung. Proses pemesanan yang terjadi pada Toko Mansure dilakukan di kasir sehingga pelanggan ketika datang tidak langsung ke tempat duduk melainkan harus ke kasir dahulu. Jika pelanggan dalam jumlah yang banyak maka akan terjadi antrian yang panjang. Selain itu, buku menu yang disediakan pun belum memiliki detail yang lengkap sehingga pelanggan akan bertanya terlebih dahulu sebelum memesan. Dengan sistem pemesanan yang berjalan saat ini, pelanggan tidak dapat melihat estimasi pembayaran yang harus dilakukan. Tidak hanya dari sisi pelanggan saja, dari sisi Toko Mansure pun dengan sistem yang ada saat ini harus mencetak 2 setruk setiap transaksi dan mencetak ulang buku menu jika terjadi perubahan sehingga membutuhkan biaya operasional yang lebih. Sistem pemesanan yang berjalan saat ini sangat memungkinkan untuk diubah sehingga tujuan dilakukannya penelitian ini adalah untuk membangun dan melakukan implementasi *back-end* pada digitalisasi pemesanan menu berbasis *website* pada Toko Mansure. Digitalisasi pemesanan dilakukan dengan menggunakan metode *extreme programming*. Metode *extreme programming* memiliki 4 tahapan yaitu perencanaan, desain, pengkodean, dan pengujian. Pembangunan *back-end* sistem dibuat menggunakan *framework* CodeIgniter dengan bahasa pemrograman PHP dan basis data MySQL.

Kata kunci: *back-end*, digitalisasi, *extreme programming*

ABSTRACT

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

By

NADIA SALSABILA

Toko Mansure is one of the cafes in Bandar Lampung City with many visitors. Having a strategic location and a building with a natural concept is one of the reasons Toko Mansure is crowded with visitors. The ordering process that occurs at Toko Mansure is carried out at the cashier so that customers when they come do not go directly to the seat but must go to the cashier first. If there are a large number of customers, there will be a long queue. In addition, the menu book provided does not have complete details so that customers will ask first before ordering. With the current ordering system, customers cannot see the estimated payment that must be made. Not only from the customer side, from the Toko Mansure side, with the current system, they have to *print* 2 receipts for each transaction and *reprint* the menu book if there are changes, which requires more operational costs. The current ordering system is very possible to be changed so that the purpose of this research is to build and implement a back-end on the digitization of website-based menu ordering at Toko Mansure. Order digitization is carried out using the extreme programming method. The extreme programming method has 4 stages, *namely* planning, design, coding, and *testing*. Back-end system development is made using the CodeIgniter framework with the PHP programming language and MySQL *database*.

Keyword: back-end, digitalization, extreme programming

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

Oleh

NADIA SALSABILA

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

Nama Mahasiswa : **Nadia Salsabila**

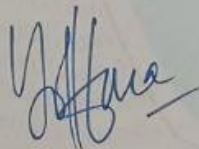
Nomor Pokok Mahasiswa : 1917051046

Program Studi : S1 Ilmu Komputer

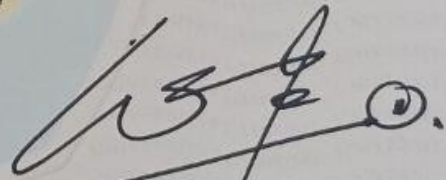
Fakultas : Matematika dan Ilmu Pengetahuan Alam

MENYETUJUI

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

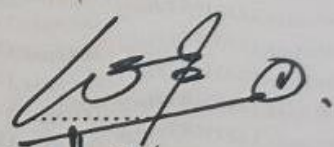


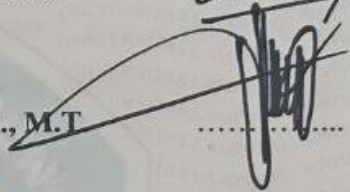
Didik Kurniawan, S.Si., M.T.
NIP. 198004192005011004

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. Eng. Heri Satria, S.Si., M.Si.
NIP. 197110012005011002

Tanggal Lulus Ujian Skripsi : 24 Juli 2023

PERNYATAAN

Saya yang bertanda tangan di bawah ini, menyatakan bahwa skripsi saya yang berjudul “**Implementasi *Back-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, 8 Agustus 2023



Nadia Salsabila
NPM 1917051046

RIWAYAT HIDUP



Penulis dilahirkan di Bandung pada tanggal 22 Januari 2001, anak dari Bapak Heri Muhammad Syapari dan Ibu Anizar, sebagai anak terakhir dari empat bersaudara.

Pendidikan sekolah dasar diselesaikan pada tahun 2013 di SDN 1 Rawa Laut. Sekolah menengah pertama diselesaikan di SMPS Al Ma'hadul Islami di Jawa Timur,

kemudian penulis melanjutkan pendidikan sekolah menengah atas di SMAS Al Ma'hadul Islami di Jawa Timur.

Pada tahun 2019 penulis melanjutkan Pendidikan di Universitas Lampung Fakultas Matematika dan Ilmu Pengetahuan Alam di Jurusan Ilmu Komputer melalui jalur SBMPTN (Seleksi Bersama Masuk Perguruan Tinggi Negeri). Berikut kegiatan yang pernah dilakukan oleh penulis selama menjadi mahasiswa.

1. Menjadi Anggota Himpunan Mahasiswa Jurusan Ilmu Komputer (HIMAKOM) di bidang Eksternal periode pada Tahun 2020 hingga 2021.
2. Menjadi Anggota Badan Eksekutif Mahasiswa Universitas (BEM U) di Bidang Kepemudaan pada tahun 2019 hingga 2020.
3. Mengikuti Karya Wisata Ilmiah FMIPA pada tahun 2019.

4. Menjadi Asisten Dosen Jurusan Ilmu Komputer pada Tahun 2020 hingga 2021
5. Mengikuti ujian sertifikasi dan mendapatkan sertifikat dari Lembaga Sertifikasi Profesi (LSP) Informatika sebagai *Junior Web Developer* oleh pada Tahun 2022.
6. Mengikuti magang di PT. Sentra Vidya Utama, Surabaya pada bulan Januari hingga Juni tahun 2022 sebagai *front-end developer*.
7. Melaksanakan Kuliah Kerja Nyata (KKN) di Kelurahan Medokan Ayu, Rungkut, Kota Surabaya pada tahun 2022.
8. Menjadi *volunteer* di Taniten sebagai *Human Resource Development* pada tahun 2023.
9. Mengikuti magang di Pro Strategic sebagai *content writer* pada tahun 2023.
10. Melaksanakan ujian sertifikasi dan mendapatkan sertifikat dari Lembaga Sertifikasi Profesi (LSP) Teknologi Digital pada bidang *digital marketing* pada tahun 2023.

MOTTO

“Karena sesungguhnya sesudah kesulitan ada kemudahan”

“Sesungguhnya sesudah kesulitan ada kemudahan”

(QS Al Insyirah: 5-6)

“Bukan kesulitan yang membuat kita takut, tapi ketakutanlah yang membuat kita sulit, karena itu jangan pernah mencoba untuk menyerah dan jangan pernah menyerah untuk mencoba. Maka jangan katakan pada Allah aku punya masalah besar, tapi katakan pada masalah aku punya Allah yang maha besar.”

(Ali bin Abi Thalib RA)

“Jangan pernah berhenti berdoa untuk yang terbaik bagi orang yang kau cintai”

(Ali bin Abi Thalib RA)

PERSEMBAHAN

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Dengan Ridho Allah Subhanahu wa Ta'ala,
teriring shalawat dan salam kepada Nabi Muhammad SAW
kupersembahkan karya sederhana ini kepada:

Bapak dan Ibuku tercinta

Heri M. Syapari dan Anizar

Kakak-kakakku tersayang

**Annisa Nurawalia, Muhammad Rijal Sidiq,
dan Muhammad Faris Akbar**

yang selalu menyayangiku dan tiada henti memberikan dukungan,
motivasi, dan doa dalam setiap langkah serta hidupku.

SANWACANA

Puji syukur kehadirat Allah Subhanahu Wa Ta'ala atas segala rahmat, karunia, dan nikmat yang tak terhingga, sehingga penulis dapat menyelesaikan laporan skripsi dengan judul “Implementasi *Back-End* Pada Digitalisasi Pemesanan Menu Makanan Dan Minuman Berbasis *Website* (Studi Kasus: Toko Mansure)” dengan baik dan tepat waktu. Tak lupa shalawat dan salam kepada Rasulullah, Nabi Muhammad Shallallahu Alaihi wasallam yang menjadi uswatun hasanah dan selalu dinantikan syafaatnya kelak di hari akhir.

Pada kesempatan ini, penulis mengucapkan terima kasih kepada Bapak dan Ibu tercinta, serta kakak-kakakku tersayang yang selalu memberikan semangat, dukungan, nasihat, dan doa yang tiada henti yang diberikan selama ini.

Tak lupa penulis juga berterimakasih kepada yang terhormat :

1. Prof. Dr. Ir. Lusmeilia Afriani, D.E.A., I.P.M., selaku Rektor Universitas Lampung.
2. Dr. Eng. Heri Satria, S.Si., M.Si. selaku Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.
3. Didik Kurniawan, S.Si., M.T. selaku Ketua Jurusan Ilmu Komputer Universitas Lampung dan Dosen Pembahas.
4. Dr. rer. Nat. Akmal Junaidi, M.Sc. selaku Sekretaris Jurusan Ilmu Komputer Universitas Lampung.
5. Yohana Tri Utami, S.Kom., M.Kom. selaku Dosen Pembimbing Utama yang selalu memberikan waktu, dukungan, dan nasihat yang tiada henti.
6. Wartariyus, S.Kom., M.T.I selaku Dosen Pembimbing Kedua yang selalu memberikan waktu, dukungan, dan nasihat yang tiada henti.

7. Pihak Toko Mansure yang telah membantu dalam menyelesaikan skripsi ini dengan baik dan lancar.
8. Bapak dan Ibu dosen Jurusan Ilmu Komputer yang telah memberikan ilmu dan pengalaman yang tiada henti kepada penulis.
9. Ibu Ade Nora Maela, Bang Zainuddin, dan Mas Naufal yang telah membantu dalam hal administrasi di Jurusan Ilmu Komputer.
10. Okta Silvi selaku rekan dalam menyelesaikan skripsi dan yang selalu ada dalam setiap keadaan.
11. Himatuy yang terdiri dari Ghina Aramitha, Lulu Agustin, Nabilah Putri, Qurrota Aini, dan Okta Silvi yang selalu ada dalam setiap keadaan.
12. Indri Mustika yang selalu ada dalam setiap keadaan.
13. Angkatan 19 Ilmu Komputer selaku teman seperjuangan yang telah memberikan pengalaman yang baik semasa kuliah.

Penulis menyadari bahwa banyak kekurangan dalam penyusunan laporan skripsi ini, sehingga diharapkan kritik dan saran yang membangun untuk pembelajaran di masa yang akan datang. Atas bantuan dari semua pihak yang telah diberikan kepada penulis dengan tulus dan ikhlas, semoga dibalas dan diridhai oleh Allah Subhanahu Wa Ta'ala. Semoga laporan ini bermanfaat bagi para pembaca.

Bandar Lampung, 24 Juli 2023

Nadia Salsabila
NPM 1917051046

DAFTAR ISI

	Halaman
DAFTAR ISI	iv
DAFTAR GAMBAR	vi
DAFTAR TABEL	ix
I PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah	3
1.3 Batasan Masalah.....	4
1.4 Tujuan Penelitian.....	4
1.5 Manfaat Penelitian.....	4
II TINJAUAN PUSTAKA	6
2.1 Uraian Landasan Teori	6
2.1.1 Aplikasi Berbasis <i>Website</i>	6
2.1.2 Digitalisasi	6
2.1.3 <i>Back-end</i>	7
2.1.4 Sistem Informasi	8
2.1.5 <i>Framework CodeIgniter</i>	9
2.1.6 Bahasa PHP (<i>Hypertext Preprocessor</i>).....	9
2.1.7 Metode <i>Extreme Programming</i>	10
2.1.8 UML (<i>Unified Modelling Language</i>).....	11
2.1.9 Apache JMeter	14
2.1.10 Selenium IDE	16
2.1.11 OWASP ZAP	17
2.1.12 Basis Data	17
2.1.13 Kamus Data.....	18
2.2 Penelitian Terdahulu.....	18
III METODOLOGI PENELITIAN	21
3.1 Waktu dan Tempat Penelitian	21
3.2 Tahapan Penelitian	21
3.2.1 Tahap Pengumpulan Data	22
3.2.2 Tahap Pengembangan Sistem	23
3.2.3 Penulisan Laporan.....	47
IV HASIL DAN PEMBAHASAN	48
4.1 Hasil.....	48
4.2 Pembahasan	48

4.2.1	Pengkodean	48
4.2.2	Pengujian.....	65
V	KESIMPULAN DAN SARAN	71
5.1	Kesimpulan.....	71
5.2	Saran.....	71
	DAFTAR PUSTAKA	73

DAFTAR GAMBAR

	Halaman
Gambar 1. Metode <i>Extreme Programming</i>	10
Gambar 2. Kardinalitas <i>One To One</i>	13
Gambar 3. Kardinalitas <i>One To Many / Many To One</i>	13
Gambar 4. Kardinalitas <i>Many To Many</i>	14
Gambar 5. Bagan Alir Tahapan Penelitian.....	22
Gambar 6. <i>Use case Diagram</i>	26
Gambar 7. <i>Entity Relation Diagram</i>	28
Gambar 8 Halaman <i>Home</i>	31
Gambar 9. Halaman Daftar Menu Pelanggan.....	31
Gambar 10. Halaman Keranjang Pesanan.....	32
Gambar 11. Halaman <i>Checkout</i>	33
Gambar 12. Halaman Daftar Pesanan.....	34
Gambar 13. Halaman Daftar Pembayaran.....	34
Gambar 14. Halaman Transaksi Pembayaran.....	35
Gambar 15. Halaman Setruk.....	35
Gambar 16. Halaman Daftar Laporan.....	36
Gambar 17. Halaman Cetak Laporan.....	37
Gambar 18. Halaman <i>Dashboard</i>	37
Gambar 19. Halaman Daftar Menu.....	38
Gambar 20. Halaman Tambah Menu.....	38
Gambar 21. Halaman Edit Menu.....	39
Gambar 22. Hapus Menu.....	39
Gambar 23. Halaman Tambah Akun dan Daftar Karyawan.....	40
Gambar 24. <i>Reset Password</i>	40
Gambar 25. Hapus Akun Karyawan.....	41

Gambar 26. Halaman <i>Login</i> Karyawan.....	41
Gambar 27. Halaman <i>Home</i>	49
Gambar 28. Potongan Kode Halaman <i>Home</i>	50
Gambar 29. Halaman Daftar Menu.....	50
Gambar 30. Potongan Kode Halaman Daftar Menu.	51
Gambar 31. Halaman Keranjang.....	51
Gambar 32. Potongan Kode Halaman Keranjang.....	52
Gambar 33. Halaman <i>Checkout</i>	52
Gambar 34. Potongan Kode Program Halaman <i>Checkout</i>	53
Gambar 35. Halaman Daftar Pembayaran.	54
Gambar 36. Potongan Kode Program Halaman Daftar Pembayaran.....	54
Gambar 37. Halaman Transaksi Pembayaran.	55
Gambar 38. Potongan Kode Halaman Transaksi Pembayaran.	55
Gambar 39. Halaman Cetak Setruk.....	55
Gambar 40. Potongan Kode Program Halaman Cetak Setruk.	56
Gambar 41. Halaman Daftar Laporan.....	56
Gambar 42. Potongan Kode Halaman Daftar Laporan.	56
Gambar 43. Halaman Cetak Laporan.....	57
Gambar 44. Potongan Kode Halaman Cetak Laporan.	57
Gambar 45. Halaman Daftar Pesanan.	58
Gambar 46. Potongan Kode Program Halaman Daftar Pesanan.....	58
Gambar 47. Halaman <i>Login</i>	59
Gambar 48. Potongan Kode Halaman <i>Login</i>	59
Gambar 49. Halaman <i>Dashboard</i>	60
Gambar 50. Potongan Kode Halaman <i>Dashboard</i>	60
Gambar 51. Halaman Daftar Menu.....	61
Gambar 52. Potongan Kode Halaman Daftar Menu.	61
Gambar 53. Halaman Tambah Menu.	61
Gambar 54. Potongan Kode Halaman Tambah Menu.	62
Gambar 55. Halaman Edit Menu.	62
Gambar 56. Potongan Kode Halaman Edit Menu.....	62
Gambar 57. Halaman Hapus Menu.....	63

Gambar 58. Potongan Kode Halaman Hapus Menu.	63
Gambar 59. Halaman Tambah Akun Dan Daftar Karyawan.	63
Gambar 60. Potongan Kode Halaman Tambah dan Daftar Karyawan.	64
Gambar 61. Hapus Akun Karyawan.	64
Gambar 62. Potongan Kode Hapus Akun Karyawan.....	64
Gambar 63. <i>Reset Password</i> Karyawan.	65
Gambar 64. Potongan Kode <i>Reset Password</i> Karyawan.	65
Gambar 65. Hasil Pengujian Performa Pelanggan.....	66
Gambar 66. Hasil <i>Test Plan</i> HTTP <i>Request</i> Karyawan	67
Gambar 67. Hasil Pengujian Integrasi	68
Gambar 68. Hasil Pemindaian Keamanan	70

DAFTAR TABEL

	Halaman
Tabel 1. <i>Use case Diagram</i>	12
Tabel 2. Kamus Data Entitas <i>User</i>	28
Tabel 3. Kamus Data Entitas Menu	28
Tabel 4. Kamus Data Entitas Pesanan	29
Tabel 5. Kamus Data Entitas Transaksi	29
Tabel 6. Kamus Data Entitas Detail_pesanan	30
Tabel 7. <i>Test Plan HTTP Request</i> Pelanggan	42
Tabel 8. <i>Test Plan HTTP Request</i> Karyawan	43
Tabel 9. Skenario Uji Integrasi	43

I PENDAHULUAN

1.1 Latar Belakang

Kafe merupakan salah satu destinasi yang paling banyak dikunjungi dalam beberapa tahun belakangan baik di kalangan remaja maupun orang dewasa. Kafe yang awalnya hanya tempat untuk menjual makanan dan minuman sekarang menjadi salah satu tempat yang sangat diminati masyarakat untuk bersantai, melepas penat setelah bekerja, mengerjakan tugas, dan foto-foto untuk memperbarui postingan di sosial media. Salah satu daya tarik pengunjung kafe adalah area yang bagus untuk berfoto. Banyak hal yang dilakukan pemilik kafe untuk membuat suasana kafe menjadi nyaman sehingga pengunjung akan merasa nyaman dan betah berlama-lama di kafe tersebut. Usaha yang dapat dilakukan adalah dari segi bangunan dan interior kafe yang unik, variasi makanan, *plating*, fasilitas yang lengkap, dan sebagainya. Salah satu fasilitas yang paling banyak dicari pengunjung adalah wifi sehingga pengunjung dapat menghemat penggunaan paket data.

Sejak Pandemi Covid-19 melanda, banyak kafe atau tempat-tempat umum yang beralih ke digital untuk memutus rantai penyebaran virus. Salah satu upaya kafe untuk memutus rantai penyebaran virus adalah dengan cara tidak memberikan buku menu kepada pengunjung untuk menghindari sentuhan fisik dengan barang-barang yang digunakan oleh banyak orang. Oleh karena itu, pemilik kafe membuat buku menu secara digital. Namun, buku menu tersebut hanya menunjukkan daftar menunya saja. Biasanya daftar menu tersebut dimasukkan ke dalam *google drive* atau dijadikan *barcode*, jika pengunjung *scan barcode* tersebut akan tertampil daftar menu dalam bentuk gambar. Proses pemesanan makanan dan minuman masih dilakukan secara manual yaitu pencatatan oleh pelayan sehingga kontak langsung

masih harus terjadi. Saat ini Indonesia telah memasuki era revolusi industri 4.0 dimana hampir semua sektor dikendalikan oleh teknologi. Teknologi dan informasi merupakan kebutuhan masyarakat saat ini. Perkembangan teknologi dan informasi mampu memberikan dampak yang besar terhadap kemajuan industri (Wahyono, 2019). Salah satu industri yang diuntungkan dengan adanya teknologi dan informasi adalah industri makanan dan minuman berbasis kafe. Perkembangan teknologi dan informasi yang sangat pesat ini diharapkan mampu membantu kafe dalam memberikan pelayanan yang memuaskan bagi pelanggan dan memudahkan dalam proses pemesanan makanan dan minuman.

Toko Mansure adalah kafe yang terletak di Jl. KH. Mas Mansur, Rawa Laut, Enggal, Kota Bandar Lampung. Proses pemesanan pada Toko Mansure masih dilakukan secara manual yaitu pelanggan akan melihat dan memesan makanan di kasir dan langsung bayar. Terdapat kekurangan pada proses pemesanan yang masih dilakukan secara manual ini, yaitu jika kafe dalam keadaan ramai pelanggan, maka pelanggan harus mengantri terlebih dahulu untuk melihat dan memesan menu. Pada daftar menu yang disediakan tidak ada detail penjelasan atau deskripsi mengenai isi dari makanan atau minuman tersebut sehingga pelanggan harus bertanya kepada pelayan dan menyebabkan antrian yang cukup lama karena pelayan harus menjelaskan mengenai menu tersebut terlebih dahulu.

Pada proses pemesanan yang terjadi saat ini di Toko Mansure, pelanggan tidak bisa melihat estimasi total harga pembelian sehingga harus menghitung secara manual sebelum memesan agar tidak melebihi *budget* yang ada. Setelah pelanggan membayar pesanan, kasir akan mencetak dua setruk yaitu untuk pelanggan dan bagian dapur sehingga membutuhkan kertas setruk yang banyak. Selain itu, proses yang dilakukan Toko Mansure ketika akan mengganti atau menambahkan menu baru adalah harus mencetak ulang buku menu yang ada untuk memperbarui menunya. Sehingga membutuhkan biaya untuk mencetak menu tersebut.

Pada tahun 2022 dengan judul penelitian “Pengelolaan Pemesanan Menu Makanan Menggunakan *Framework Flask Python*” pernah memberikan solusi supaya pelanggan dapat memesan menu dari tempat duduk saja (Sholeh, Aji, Riady, & Qasthari, 2022). Namun, solusi tersebut hanya sampai di pelanggan saja. Tidak sampai ke bagian dapur dan kasir. Selain itu, pada penelitian tersebut belum ada

daftar menu yang menunjukkan menu rekomendasi dan laporan transaksi. Sehingga solusi yang diberikan oleh peneliti tersebut masih perlu dikembangkan lagi.

Berdasarkan latar belakang tersebut, pada penelitian ini dibuat sistem pemesanan menu berbasis *website* dengan menggunakan *framework* CodeIgniter versi 4. Pemesanan menu yang telah beralih ke digital dapat memudahkan pelanggan dalam memesan menu tanpa harus mengantri karena pelanggan akan memesan menu dari meja yang ditempati dan dapat melihat estimasi pembayaran yang harus dibayarkan. Sistem pemesanan menu ini tidak hanya memudahkan pelanggan saja melainkan dapat mengurangi biaya operasional dalam mencetak menu dan cukup cetak setruk untuk pelanggan saja. Selain itu sistem ini dapat membantu dalam hal pengelolaan kafe dan meringankan pekerjaan karyawan khususnya bagian koki dan kasir karena setelah pelanggan memesan maka pesanan akan terkirim ke bagian koki kemudian ke kasir. Kasir tidak perlu mendatangi karyawan bagian dapur untuk memberikan pesanan pelanggan karena koki dapat melihat sendiri daftar pesannya.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dijabarkan di atas, rumusan masalah pada penelitian ini adalah sebagai berikut.

1. Bagaimana membangun sebuah sistem informasi yang dapat memudahkan proses pemesanan makanan dan minuman pada Toko Mansure?
2. Bagaimana membangun sebuah sistem informasi yang dapat menampilkan tagihan pembayaran pelanggan secara otomatis?
3. Bagaimana membangun sebuah sistem informasi yang dapat meringankan pekerjaan karyawan?

1.3 Batasan Masalah

Berdasarkan latar belakang yang telah dijabarkan di atas, maka penelitian ini dibatasi pada:

1. Fokus penelitian ini pada bagian *back-end* dari pengembangan sistem informasi pemesanan menu berbasis *website*.
2. Sistem ini hanya dapat digunakan di Toko Mansure.
3. Sistem ini dibangun menyesuaikan permintaan dan kebutuhan dari pihak Toko Mansure.
4. Sistem ini dapat digunakan oleh karyawan Toko Mansure yang terdaftar pada sistem.

1.4 Tujuan Penelitian

Berdasarkan latar belakang yang telah dijabarkan di atas, maka tujuan dari penelitian ini adalah membangun dan melakukan implementasi *back-end* pada sebuah sistem informasi pemesanan menu berbasis *website* pada Toko Mansure.

1.5 Manfaat Penelitian

Berdasarkan latar belakang yang telah dijabarkan di atas, manfaat pada penelitian ini adalah sebagai berikut.

- A. Pelanggan Toko Mansure
 1. Memudahkan pelanggan dalam memesan makanan dan minuman sehingga tidak perlu mengantri.
 2. Memudahkan pelanggan dalam mengetahui detail atau deskripsi makanan yang ada pada daftar menu.
 3. Memudahkan pelanggan dalam mengetahui estimasi pengeluaran yang harus dilakukan untuk membayar pesanan.

B. Toko Mansure

1. Mengurangi biaya operasional kafe seperti buku menu dan bukti pesanan yang akan diberikan ke bagian dapur.
2. Meringankan pekerjaan karyawan Toko Mansure.
3. Memudahkan Toko Mansure dalam mengelola laporan transaksi.
4. Mampu meningkatkan kualitas dan daya saing kafe.

II TINJAUAN PUSTAKA

2.1 Uraian Landasan Teori

2.1.1 Aplikasi Berbasis *Website*

Menurut (Ma'rifat & Kesuma, 2018) aplikasi berbasis *website* adalah *software client-server* yang berjalan di web *browser*. Aplikasi berbasis web mampu memudahkan *developer* dalam mengembangkan *software* karena dapat dijalankan di berbagai *platform* sistem operasi karena *software* tersebut berjalan di web *browser* sehingga sistem operasi Windows, Linux, dan Mac bisa menggunakannya. Aplikasi berbasis *website* tidak membutuhkan perangkat tambahan karena untuk mengaksesnya hanya membutuhkan internet dan web *browser* saja. Aplikasi berbasis web adalah aplikasi yang bisa diakses dengan internet atau intranet (Susanti, Junianto, & Rachman, 2017).

2.1.2 Digitalisasi

Digitalisasi adalah proses mengubah sesuatu yang berbentuk fisik seperti buku, peta, dan bentuk kertas lainnya menjadi virtual dan digital (Siregar, 2019). Saat ini berkas-berkas yang masih berbentuk fisik tidak lagi disimpan didalam lemari melainkan dengan adanya digitalisasi, berkas tersebut dapat disimpan di server sehingga dapat diakses oleh siapapun dan dimanapun hanya dengan perangkat elektronik.

Menurut (Asaniyah, 2017), digitalisasi adalah proses perubahan dari bentuk tercetak atau fisik menjadi elektronik.

2.1.3 *Back-end*

Menurut (Arhandi, 2016), *back-end* adalah tempat untuk memproses suatu sistem yang berjalan dimana, pada *back-end* ini data dapat diproses, ditambahkan, diubah atau dihapus. *Back-end* sering juga disebut sebagai *server side*. Proses yang tidak dilihat atau yang tidak berinteraksi langsung dengan *user* berada pada *back-end* seperti *database*, server, keamanan, dan sebagainya. Menurut (Kurniawan, Humaira, & Rozi, 2020), *back-end* adalah bagian dibalik layar dari suatu *software*. Bahasa pemrograman yang digunakan oleh seorang *back-end developer* adalah PHP, ruby, python, dan bahasa lainnya.

Back-end bertujuan untuk mendukung *front-end* agar dapat bekerja sebagaimana mestinya. Seorang *back-end developer* dibutuhkan dalam pengembangan suatu sistem atau aplikasi yang mempunyai data yang selalu berubah-ubah. Terdapat beberapa kriteria atau ciri-ciri pada *back-end software* yang baik.

1. Memiliki performa yang baik

Back-end yang berkualitas harus memiliki performa yang baik untuk mencapai kinerja yang baik pula. Untuk mengetahui apakah performa yang dimiliki baik atau tidak, maka harus dilakukan pengujian performa atau kinerja sistem. Menurut (Azura, 2023) pengujian performa bertujuan untuk melihat apakah sistem dapat menangani jumlah *user*, lalu lintas, dan permintaan data yang tinggi tanpa mengurangi performa *software*. *Tools* yang dapat digunakan untuk menguji performa sistem adalah Apache JMeter dengan melakukan simulasi beban tinggi pada sistem kemudian menganalisis hasilnya. *Tools* Apache JMeter mampu melakukan pengujian pada HTTP (*Hypertext Transfer Protocol*), basis data, FTP (*File Transfer Protocol*), dan protokol lainnya.

2. Memiliki keamanan yang baik

Salah satu kriteria pada *back-end* yang baik adalah memiliki keamanan sistem yang baik agar terhindar dari serangan *hacker* dan tidak ada pencurian

data. Keamanan sebuah sistem harus diuji terlebih dahulu sebelum benar-benar digunakan oleh *user* untuk mengetahui kerentanan keamanan sistem tersebut. Menurut (Nader Y. , 2023) dalam artikel yang diunggahnya pada sebuah *website*, *tools* OWASP (*Open Web Application Security Project*) ZAP (*Zed Attack Proxy*) adalah salah satu alat teratas untuk melakukan pengujian keamanan. OWASP ZAP adalah *security testing tools* yang *open source* dengan tujuan untuk menemukan kerentanan keamanan pada sistem.

3. Mudah diintegrasikan dengan sistem lain

Back-end yang baik harus mudah dihubungkan atau diintegrasikan dengan sistem lainnya seperti *front-end*, *database*, dan lainnya agar dapat berjalan sebagaimana mestinya (Lawrence, 2021). Interaksi antar komponen tersebut harus diuji untuk mengetahui bahwa interaksi antar komponen tersebut berjalan dan berfungsi dengan baik. Salah satu alat yang dapat digunakan untuk melihat integrasi pada sistem tersebut berjalan dengan baik adalah Selenium IDE. Alat Selenium IDE adalah pengujian *automation* yang sering digunakan untuk integrasi *testing* dan dapat digunakan untuk *back-end testing* melalui API (*Application Programming Interface*) atau interaksi langsung dengan unsur-unsur web lainnya (Azura, 2023). Menurut (Hamilton, 2023) tahapan pada saat pengujian integrasi adalah sebagai berikut.

- a. Membuat rencana pengujian
- b. Membuat skenario uji
- c. Melakukan pengujian
- d. Menganalisis hasil pengujian

Menurut (Nanda, 2021) dalam artikelnya, proses pada pengujian *back-end* adalah menyiapkan lingkungan pengujian, membuat *test scenario*, melakukan pengujian, melakukan analisis hasil pengujian, dan membuat laporan.

2.1.4 Sistem Informasi

Sistem informasi adalah suatu kegiatan untuk mengumpulkan data dan mengolahnya agar dapat mencapai tujuan yang diinginkan (Wijaya, Hendrastuty, Damayanti, & An'ars, 2022). Sistem informasi merupakan sebuah sistem yang ada

di dalam suatu organisasi untuk mendukung operasi, bersifat manajerial, dan untuk mengambil suatu keputusan berdasarkan laporan-laporan yang disediakan. Menurut (Novitasari, Adrian, & Kurnia, 2021), sistem informasi merupakan sistem yang dapat memberikan keluaran berupa informasi yang berguna untuk level manajemen.

2.1.5 Framework CodeIgniter

Menurut (Sofiani & Nurhidayat, 2019) *framework* CodeIgniter adalah *framework* PHP yang dapat mempercepat pengembangan *software* untuk membuat aplikasi berbasis *website* yang dilengkapi *library* dan *helper* untuk mempermudah proses pembuatan. *Framework* CodeIgniter adalah aplikasi *open source* dengan model MVC (*Model, View, Controller*) untuk memudahkan pengembang dalam membangun sebuah sistem karena struktur kode yang dihasilkan menjadi lebih terstruktur dan mempunyai standar. Namun, *Framework* CodeIgniter masih memberikan kebebasan kepada *developer* untuk menulis kode-kode tertentu dengan menggunakan *syntax* umum pada PHP sehingga tidak mengharuskan pengembangan untuk mengikuti aturan penulisan pada CodeIgniter. *Framework* CodeIgniter memiliki dokumentasi yang lengkap karena CodeIgniter menyediakan panduan penggunaan yang lengkap dan jelas (Sallaby & Kanedi, 2020).

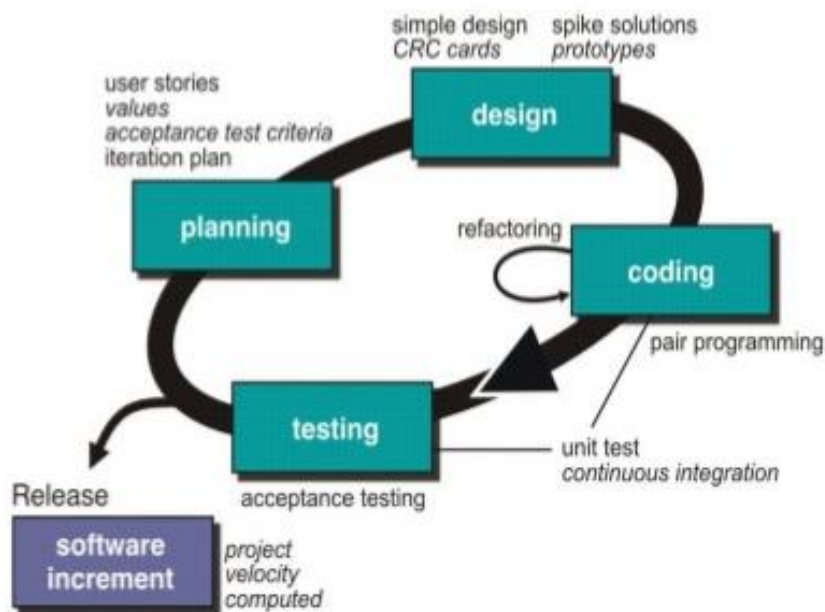
2.1.6 Bahasa PHP (*Hypertext Preprocessor*)

Bahasa pemrograman PHP (*Hypertext Preprocessor*) adalah Bahasa pelengkap HTML (*Hypertext Markup Language*) yang memiliki pengolahan data dan pemrosesan data. Bahasa pemrograman PHP adalah bahasa yang berbentuk *script* dan terkenal dengan bahasa *scripting* yang dijalankan di server sedangkan yang dikirimkan ke *browser* adalah hasilnya. Bahasa pemrograman PHP digunakan untuk membuat aplikasi berbasis *website* dan merupakan bahasa *open source* yang dapat dijalankan dengan berbagai sistem operasi seperti Windows, Linux, atau Macintosh (Wibowo, Kanedi, & Jumadi, 2015). Menurut (Erdani, Aditia, Rodiah,

Ciptyasih, & Santi, 2019), bahasa pemrograman PHP adalah bahasa yang dapat disatukan dengan berbagai macam *database*.

2.1.7 Metode *Extreme Programming*

Extreme programming adalah metode pengembangan sistem yang bertujuan untuk meningkatkan kualitas sistem terhadap perubahan dan kebutuhan pelanggan secara cepat (Ariyanti, Satria, & Alita, 2020). Metode *extreme programming* berfokus pada pendekatan berorientasi objek serta tim dengan skala kecil hingga menengah. Menurut (Budiarti, 2020), metode *extreme programming* adalah metode pengembangan sistem yang dapat menyederhanakan berbagai macam tahapan pengembangan menjadi lebih efisien, adaptif, dan fleksibel. Tahapan metode *extreme programming* dapat dilihat pada Gambar 1.



Gambar 1. Metode *Extreme Programming* (Budiarti, 2020)

Penjelasan untuk setiap tahapan pada metode *extreme programming* adalah sebagai berikut.

1. Perencanaan

Perencanaan adalah tahapan awal dalam membangun sebuah sistem. Pada tahapan perencanaan yang dilakukan adalah menentukan permasalahan yang akan diselesaikan, menentukan kebutuhan sistem, dan menentukan pelaksanaan pembuatan sistem.

2. Desain

Tahap kedua yang harus dilakukan adalah desain. Tahapan desain mencakup perancangan sistem, *database*, dan arsitektur. Pemodelan sistem dan arsitektur menggunakan UML (*Unified Modelling Language*) dan perancangan *database* menggunakan ERD.

3. Pengkodean

Tahap ketiga dari model *extreme programming* adalah implementasi desain yang telah dibuat dengan bahasa pemrograman. Bahasa pemrograman yang digunakan adalah bahasa PHP dan untuk manajemen basis data menggunakan MySQL.

4. Pengujian

Tahap terakhir dari metode *extreme programming* adalah pengujian kualitas sistem untuk mengetahui kesalahan apa saja yang terjadi ketika sistem dijalankan dan untuk mengetahui apakah sistem sudah sesuai dengan kebutuhan pelanggan atau tidak.


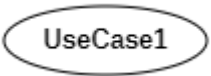


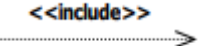

2.1.8 UML (*Unified Modelling Language*)

UML (*Unified Modelling Language*) adalah bahasa yang digunakan untuk mendefinisikan kebutuhan, membuat analisis, dan desain dalam mengembangkan sistem berorientasi objek (Hendini, 2016). Menurut (Sukmawati & Priyadi, 2019), UML (*Unified Modelling Language*) dapat membantu menggambarkan, menetapkan, menciptakan, dan menyimpan artefak sistem secara efektif. UML (*Unified Modelling Language*) digunakan sebagai alat pendukung pembuatan suatu *software* atau sistem sehingga memudahkan *developer* dalam merencanakan, menggambarkan, dan dokumentasi dalam pembuatan sistem sebelum proses implementasi dilakukan.

1. Use Case Diagram

Use case diagram adalah penggambaran fungsionalitas dari suatu sistem berdasarkan sudut pandang *user* atau pengguna. *Use case* menggambarkan proses apa saja yang akan dilakukan oleh sistem dan apa saja yang bisa dilakukan oleh pengguna (Astuti, 2009). Penggunaan *use case diagram* dapat membantu *developer* dalam membuat kebutuhan sistem, menjelaskan kepada klien mengenai sistem tersebut, dan dapat menentukan siapa saja yang akan menggunakan sistem tersebut dan apa saja yang dapat dilakukan oleh sistem. Pada *use case diagram* terdapat berbagai macam simbol untuk mendefinisikan sesuatu. Simbol-simbol pada *use case diagram* dapat dilihat pada Tabel 1.

Tabel 1. *Use case Diagram*

No.	Simbol	Nama	Keterangan
1.		Aktor	Peran yang digunakan oleh <i>user</i> untuk berinteraksi dengan sistem atau pengguna sistem.
2.		<i>Use Case</i>	Fungsionalitas yang disediakan oleh sistem atau aksi yang akan ditampilkan oleh sistem.
3.		<i>Association</i>	Penghubung antara <i>actor</i> dengan <i>use case</i> atau <i>use case</i> dengan <i>use case</i> lainnya.
4.		<i>Generalization</i>	Hubungan antara <i>use case</i> yang umum dengan khusus.
5.		<i>Include</i>	Relasi <i>use case</i> tambahan sebagai syarat dijalankannya <i>use case</i> tersebut.
6.		<i>Extend</i>	Relasi <i>use case</i> tambahan ke <i>use case</i> lainnya yang dapat berdiri sendiri walaupun tanpa <i>use case</i> tambahan tersebut.

2. Entity relationship diagram (ERD)

Menurut (Rahmayu, 2016), *entity relationship diagram* atau ERD adalah pemodelan data yang digunakan untuk mendesain *database*. Komponen dasar pada *entity relationship diagram* adalah relasi atau hubungan, atribut, dan entitas. Penjelasan pada setiap komponen dasar adalah sebagai berikut.

a) Relasi

Hubungan antar entitas. Pada relasi terdapat derajat relasi yang menunjukkan jumlah entitas yang berperan dalam satu relasi. Terdapat beberapa jenis derajat relasi salah satunya adalah kardinalitas. Kardinalitas adalah jumlah relasi antar entitas. Kardinalitas menunjukkan jumlah kemungkinan kemunculan suatu entitas yang dapat direlasikan dengan entitas lainnya. Jenis kardinalitas pada ERD adalah sebagai berikut.

A. *One to One*

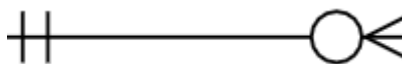
One to one adalah relasi yang menunjukkan jumlah kemungkinan entitas pertama adalah sebanyak satu kali terhadap entitas kedua. Kardinalitas *one to one* dapat dilihat pada Gambar 2.



Gambar 2. Kardinalitas *One To One*.

B. *One to Many / Many to One*

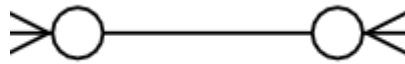
One to many adalah jumlah kemungkinan hubungan pada entitas pertama adalah satu dan memiliki jumlah hubungan yang banyak atau lebih dari satu pada entitas kedua. *Many to one* kebalikan dari *one to many* yaitu jumlah kemungkinan hubungan pada entitas pertama adalah banyak dan entitas kedua adalah satu. Hubungan antara entitas pertama dengan kedua memiliki hubungan lebih dari satu. Kardinalitas *one to many* atau *many to one* dapat dilihat pada Gambar 3.



Gambar 3. Kardinalitas *One To Many / Many To One*.

C. *Many to Many*

Many to many adalah jumlah kemungkinan hubungan pada entitas pertama dengan entitas kedua adalah lebih dari satu dan begitupun sebaliknya, jumlah kemungkinan entitas kedua dengan entitas pertama adalah lebih dari satu. Kardinalitas *many to many* dapat dilihat pada Gambar 4.



Gambar 4. Kardinalitas *Many To Many*.

b) Atribut

Atribut adalah suatu karakteristik dari *entity* yang dapat dideskripsikan. Atribut memiliki dua jenis yaitu *identifier* dan *descriptor*. *Identifier* atau *key* adalah penentu utama entitas dan harus unik karena tidak boleh sama dengan lainnya. *Identifier* merupakan pembeda antar entitas. *Descriptor* atau *non key attribute* adalah untuk merincikan karakteristik dari entitas yang tidak unik.

c) Entitas

Entitas atau *entity* adalah kumpulan objek yang dapat didefinisikan sehingga dapat dibedakan dengan objek lainnya. Entitas yang ada di ERD dapat digambarkan dengan bentuk persegi panjang.

2.1.9 Apache JMeter

Apache JMeter adalah alat *software open source* yang berfungsi untuk menghitung kinerja sebuah sistem dengan mensimulasikan beban kerja dan *user* sistem sehingga dapat terlihat kinerja atau performa dari sistem tersebut. Apache JMeter adalah alat pengujian kinerja yang paling disukai di dunia, bahkan perusahaan besar seperti Google, Facebook, dan perusahaan publik atau swasta menggunakan Apache JMeter untuk memeriksa apakah sistemnya dapat digunakan dengan baik atau tidak. Apache JMeter mendukung berbagai server dan protokol seperti HTTP, HTTPS, SOAP, dan sebagainya. Mampu melakukan pengujian beban dan kinerja dengan berbagai server dan *browser* seperti *website*. Sebelum *install* Apache JMeter diharuskan melakukan instalasi java terlebih dahulu. Tiga tahapan untuk melakukan pengujian performa adalah sebagai berikut.

1. Menambahkan utas

Pada bagian utas terdapat *threads users*, *ramp-up periode*, dan *loop count*. Menurut (Khader, 2022) *threads users* adalah seberapa banyak pengguna yang akan disimulasikan untuk menjalankan sistem. *Ramp-up period* adalah berapa

banyak waktu yang dibutuhkan pengguna dalam menjalankan skenario. Rumus pada *ramp-up period* adalah $\frac{\text{ramp-up period}}{\text{threads user}}$ dalam hitungan detik atau *second*. Jika terdapat 10 *threads* dan 100 *ramp-up period* maka, terdapat 10 detik untuk 1 *threads* sehingga untuk 10 *threads* memiliki waktu 100 detik. *Loop count* adalah jumlah putaran untuk setiap pengguna yang menunjukkan berapa banyak setiap tugas akan diselesaikan oleh *user*. Rumus pada *loop count* adalah $\text{threads user} \times \text{loop count}$. Jika terdapat 2 *threads* dan 3 *loop count*, maka skenario yang sama akan dilakukan sebanyak 6 kali.

2. Menambahkan *sampler*

Pada bagian *sampler* adalah untuk menentukan tugas yang akan dilakukan. Cara untuk menambahkan *sampler* adalah dengan menekan tombol kanan *mouse* kemudian pilih *add->sampler->HTTP request*.

3. Menambahkan *listener*

Tahapan yang terakhir adalah menambahkan *listener* dengan tujuan untuk memberikan hasil dari pengujian yang dapat berupa grafik, tabel, dan sebagainya (Shethi, 2022). Menurut (Lumbantobing, 2021) salah satu *listener* yang ada adalah *summary report* dalam bentuk tabel dengan *field* sebagai berikut.

a. Label

Label merupakan nama pada *HTTP request* yang dijalankan.

b. *Samples*

Samples adalah jumlah *user per request* yang didapatkan dari $\text{threads user} \times \text{loop count}$ per label.

c. *Average*

Average adalah waktu rata-rata yang dibutuhkan untuk menjalankan setiap label dalam satuan milidetik.

d. Min

Min adalah minimal waktu yang dibutuhkan untuk menjalankan setiap label dalam satuan waktu milidetik. Total nilai min didapatkan dari waktu tersingkat pada label yang tersedia.

- e. *Max*
Max adalah maksimal waktu yang dibutuhkan untuk menjalankan setiap label dalam satuan waktu milidetik. Total nilai *max* didapatkan dari waktu terlama pada label yang tersedia.
- f. *Std. dev*
Std. dev adalah penyebaran kumpulan data relatif terhadap rata-ratanya. Semakin kecil nilai *Std. dev*, maka datanya semakin konsisten. Nilai *Std. dev* pada setiap label sebaiknya lebih kecil atau sama dengan setengah dari nilai rata-rata dari setiap label.
- g. *Error %*
Menunjukkan jumlah kecacatan pada setiap label dalam satuan persen.
- h. *Throughput*
Throughput adalah waktu jumlah request yang berhasil diproses oleh server per unit waktu yaitu jam, menit, dan detik. Waktu tersebut dikalkulasikan dari awal sample hingga akhir sample. Semakin tinggi nilai *Throughput*, maka akan semakin baik.
- i. *Received KB/sec*
Received KB/sec adalah jumlah data yang berhasil diunduh oleh server selama proses *performance testing* dalam satuan *kilobyte* per detik.
- j. *Sent KB/sec*
Sent KB/sec adalah jumlah data yang berhasil dikirim dari server selama proses *performance testing* dalam satuan *kilobyte* per detik.
- k. *Avg bytes*
Avg bytes adalah rata-rata *byte* yang berhasil diunduh oleh server.

2.1.10 Selenium IDE

Selenium IDE adalah alat rekam yang digunakan untuk melakukan pengujian. Sebelum memulainya, ekstensi selenium IDE harus ditambahkan terlebih dahulu pada *browser* yang akan digunakan kemudian membuat kasus uji. Masukkan nama *project* dan URL pada kolom yang disediakan. Proses kerja selenium IDE adalah dengan merekam seluruh kegiatan pada sistem. Terdapat tiga

kategori utama yaitu *command*, *target*, dan *value* (Unadkat, 2023). Tahapan pengujian yang harus dilakukan jika menggunakan selenium IDE adalah membuat projek baru kemudian membuat kasus uji. Ketika membuat kasus uji akan diminta untuk memasukkan URL. Setelah memasukkan URL, tekan tombol “*rec*” untuk merekam aktivitas pada sistem. Setelah menyelesaikan kasus uji kembali ke selenium IDE untuk memberhentikan perekaman dan kemudian jalankan pengujianya. Jika pengujian sudah dilakukan, maka akan tertampil log pengujian pada setiap *test case* (Walters, 2022).

2.1.11 Basis Data

Basis data adalah kumpulan data yang disimpan di dalam komputer secara sistematis sehingga dapat dikelola untuk menghasilkan informasi. Konsep dasar pada basis data adalah kumpulan data yang saling berhubungan satu sama lain yang dirangkai berdasarkan skema tertentu. Hubungan antar data tersebut menggunakan *key* dari tiap tabel. Model pada basis data yang umum digunakan adalah model relasional yaitu, kumpulan data dalam bentuk tabel. Setiap tabel terdiri dari baris dan kolom (ANDARU, 2018). Menurut (Haryanto & Argadila, 2019), basis data adalah gabungan dari dua kata yaitu basis dan data. Basis adalah suatu tempat untuk bersatu atau berkelompok sedangkan data adalah sekumpulan informasi yang berupa fakta untuk mewakili suatu objek.

2.1.12 OWASP ZAP

Menurut (Kushchi, 2022) pada artikel yang ditulisnya, OWASP ZAP (Zed Attack Proxy) adalah alat dengan sumber terbuka yang berfungsi untuk menemukan kerentanan keamanan pada suatu *software*, memahami dan memperbaiki masalah keamanan, dan menjaga kebersihan keamanan dalam waktu yang lama. ZAP tersedia untuk sistem operasi Windows, Linux, dan MacOS. Jika ingin melihat kerentanan pada keamanan suatu sistem informasi, maka dapat dengan melakukan pemindaian otomatis. Pengujian dilakukan dengan memasukkan URL (*Uniform Resource Locator*) pada kolom yang disediakan dan menekan tombol *attack*.

Kemudian hasil pengujian yang berupa daftar kerentanan keamanan sistem akan tampil dengan simbol *alert*.

Menurut (Hristovski & Kotevski, 2021) OWASP ZAP adalah alat dengan sumber terbuka yang dapat dilakukan untuk pengujian penetrasi di bawah OWASP (Open Wes Application Security Project). Pemindaian ZAP disebut laba-laba. Ketika tombol *attack* ditekan, maka ZAP akan merayapi sistem dengan laba-labanya dan akan memindai seluruh halaman yang ditemuinya kemudian akan menyerang seluruh halaman, fungsionalitas, dan parameternya untuk menemukan kerentanan pada sistem. Kategori risiko pada hasil pemindaian adalah dengan warna merah, oren, kuning, biru, dan hijau. Warna merah menunjukkan sistem yang dipindai memiliki risiko yang tinggi, oren adalah menengah, kuning rendah, biru adalah informasi, dan hijau adalah positif.

2.1.13 Kamus Data

Kamus data adalah penjelasan informasi mengenai data-data dan kebutuhan yang ada di dalam *database*. Penggunaan kamus data sangat dibutuhkan oleh *developer* karena dengan adanya kamus data *developer* dapat mengetahui mengenai struktur pemakaian data, dapat mengetahui inputan dan outputan data, dan dapat menghindari penggunaan data yang sama. Dengan menggunakan kamus data, *developer* dapat mengorganisasikan seluruh data yang digunakan oleh perangkat lunak (Utomo & Suryana, 2022). Menurut (Haryanto & Argadila, 2019), kamus data adalah penjelasan mengenai perputaran data yang ada di dalam sistem.

2.2 Penelitian Terdahulu

Terdapat empat penelitian terdahulu yang digunakan sebagai referensi dalam penelitian ini.

a) Sistem Informasi Pemesanan Menu Makanan Pada Restoran Eighteen Pies Berbasis Web.

Restoran eighteen pies adalah unit usaha yang bergerak dibidang makanan. Mekanisme pemesanan yang terjadi di restoran eighteen pies adalah pelanggan

datang ke restoran kemudian pelayan akan memberikan buku menu dan proses pencatatan pesanan pelanggan menggunakan kertas kemudian pesanan tersebut diinputkan pada *point of sales* dan pelanggan akan membayar pesanan di kasir.

Metode pengembangan yang dilakukan adalah dengan metode *waterfal*. Pengguna dalam aplikasi ini adalah pelanggan, pelayan, admin, dan kasir. Proses pada aplikasi pemesanan menu makanan pada restoran eighteen pies adalah pelanggan harus *register* dan *login* terlebih dahulu sebelum memesan makanan, proses pembayaran dilakukan di kasir dan terdapat laporan keuangan.

Namun, pada penelitian tersebut belum terdapat fitur yang dapat menampilkan daftar menu yang di rekomendasikan dan belum terdapat fitur yang dapat memudahkan karyawan bagian dapur seperti koki. Koki masih harus mendapat informasi pesanan secara manual yaitu pelayan yang akan memberikan informasi mengenai pesanan pelanggan kepada koki (Dendi & Nurjaya, 2022).

b) Pengelolaan Pemesanan Menu Makanan Menggunakan *Framework Flask Python*.

Pemesanan menu makanan pada penelitian ini menggunakan *framework flask python* dan menggunakan metode *waterfall*. Pada aplikasi ini terdapat dua pengguna yaitu admin dan *user* atau pelanggan. Proses yang dapat dilakukan oleh admin adalah *login* sistem, mengelola daftar menu, melihat daftar menu dan pesanan, dan mengelola pemesanan. Proses yang dapat dilakukan oleh pelanggan adalah mengelola pemesanan dan melihat daftar menu.

Sistem pemesanan menu makanan belum sampai pada proses dapur dan kasir. Proses dapur dan kasir masih dilakukan secara manual, hanya proses pemesanan dan pengelolaan daftar menu saja yang sudah terdigitalisasi. Oleh karena itu tidak ada fitur laporan transaksi pembayaran pada sistem tersebut dan belum terdapat fitur yang dapat menampilkan daftar menu yang direkomendasikan oleh kafe (Sholeh, Aji, Riady, & Qasthari, 2022).

c) Rancang Bangun Sistem Informasi Pemesanan Makanan Berbasis Web Pada Cafe Surabiku.

Kafe surabiku adalah unit usaha yang bergerak dibidang makanan. Kafe surabiku mengalami kesulitan dalam melayani pemesanan karena masih menerapkan sistem secara manual dan sistem pencatatan masih dilakukan secara manual atau belum terkomputerisasi. Pelanggan harus menunggu pelayan terlebih

dahulu untuk dilayani dan pelayan harus mencatat pesanan menggunakan kertas sehingga tidak efisien dalam segi waktu.

Pembuatan aplikasi pemesanan ini menggunakan Bahasa pemrograman PHP dan *database* MySQL. Metode pengembangan aplikasi yang dilakukan pada penelitian tersebut menggunakan metode *waterfall*. Aplikasi ini terdapat tiga *user* yaitu admin, pembeli, dan kasir. Proses yang dapat dilakukan pada aplikasi ini adalah pelanggan harus *login* terlebih dahulu dengan nama dan nomor meja, dapat memesan makanan dan minuman, pembayaran dilakukan di kasir.

Aplikasi pemesanan yang ada di kafe surabiku belum sampai pada proses dapur dan belum terdapat fitur yang dapat menampilkan daftar menu yang rekomendasi. Karyawan bagian dapur atau koki akan diberikan informasi mengenai pesanan pelanggan oleh pelayan lain, koki tidak dapat melihat langsung pesanan pelanggan tanpa harus menerima informasi dari pelayan lainnya (Saputri, Oktavia, Ramdhani, & Suherman, 2019).

d) Aplikasi Pemesanan Makanan Berbasis Web Di Ramen Shinju -Ranju-.

Ramen shinju merupakan kedai ramen yang menjadi salah satu destinasi mahasiswa universitas Telkom dan masyarakat sekitar untuk menikmati makanan khas jepang yaitu ramen. Pengunjung kedai ramen shinju tidak jarang harus *waiting list* karena padat pengunjung. Proses pemesanan dan pencatatan pesanan yang ada di kedai ramen shinju masih secara manual yaitu menggunakan kertas.

Dalam pengembangan aplikasi pemesanan makanan berbasis web di kedai ramen shinju digunakan metode *waterfall* dengan menggunakan *framework* CodeIgniter dan *database* MySQL. Pengguna aplikasi tersebut adalah pelanggan dan admin dengan proses yaitu sebelum memesan makanan pelanggan harus registrasi terlebih dahulu dan proses pembayaran ada di kasir.

Namun, pada aplikasi pemesanan yang ada di ramen shinju belum terdapat fitur yang dapat menampilkan daftar menu yang rekomendasi dan belum sampai pada proses dapur. Proses dapur masih dilakukan secara manual dan terdapat saran yaitu penambahan fitur laporan transaksi untuk melihat data rekapan transaksi yang terjadi (Setiyani, 2019).

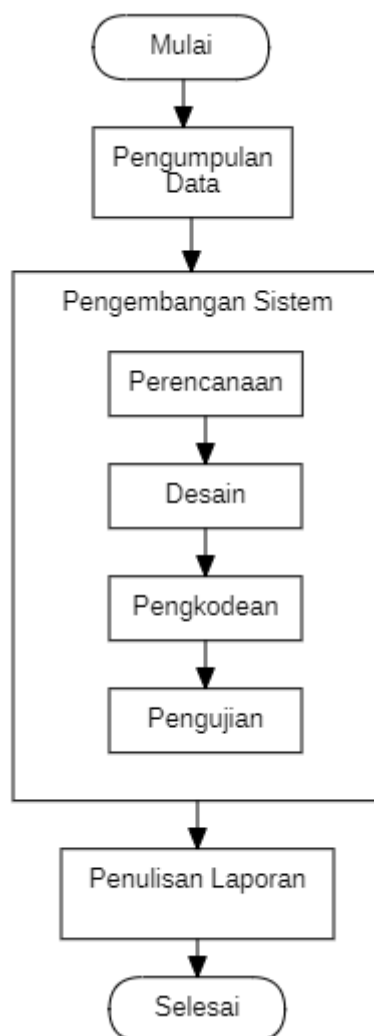
III METODOLOGI PENELITIAN

3.1 Waktu dan Tempat Penelitian

Penelitian dilakukan pada semester genap tahun ajaran 2022/2023 di Gedung Ilmu Komputer Jurusan Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung dan Toko Mansure yang beralamat di Jl. KH. Mas Mansur No.74, Rw. Laut, Enggal, Kota Bandar Lampung.

3.2 Tahapan Penelitian

Tahapan penelitian yang dilakukan terdapat tiga bagian yaitu pengumpulan data, pengembangan sistem, dan penulisan laporan. Bagan tahapan penelitian dapat dilihat pada Gambar 5.



Gambar 5. Bagan Alir Tahapan Penelitian.

3.2.1 Tahap Pengumpulan Data

Tahapan pengumpulan data didapatkan dari wawancara, studi pustaka, dan observasi. Pengumpulan data dilakukan untuk mengetahui bagaimana alur proses yang terjadi di Toko Mansure dan untuk memahami teori yang dilakukan pada penelitian ini. Tahapan pengumpulan data sebagai berikut.

1. Wawancara

Wawancara dilakukan kepada pemilik Toko Mansure dan karyawan yang bekerja di Toko Mansure. Proses wawancara dilakukan untuk mengetahui bagaimana proses pemesanan, pencatatan, proses yang dilakukan oleh kasir dan

koki. Dengan proses wawancara didapatkan informasi mengenai kebutuhan dalam pembuatan sistem pemesanan menu.

2. Studi Pustaka

Proses studi pustaka dilakukan untuk mendapatkan informasi mengenai landasan teori yang digunakan dalam penelitian. Studi pustaka adalah metode yang dilakukan untuk mengumpulkan data pendukung dalam proses pembuatan sistem pemesanan menu berbasis web. Data-data yang didapatkan berasal dari literatur yang dapat dipertanggung jawabkan seperti jurnal dan artikel yang berkaitan dengan penelitian ini.

3. Observasi

Pengumpulan data dengan observasi dilakukan untuk mengetahui permasalahan yang sedang terjadi di Toko Mansure dan proses pemesanan yang terjadi. Observasi dilakukan dengan mengamati pelanggan yang datang untuk memesan di Toko Mansure.

3.2.2 Tahap Pengembangan Sistem

Tahap selanjutnya adalah pengembangan sistem yang menggunakan metode *extreme programming*. Tahapan yang dilakukan pada metode *extreme programming* adalah perencanaan, desain, pengkodean, dan pengujian.

1. Perencanaan

Pada tahap perencanaan yang dilakukan adalah menentukan permasalahan, analisis kebutuhan sistem, dan menentukan pelaksanaan pengembangan suatu sistem. Penjelasan mengenai tahapan yang dilakukan pada bagian perencanaan adalah sebagai berikut.

A. Kebutuhan Fungsional

Kebutuhan fungsional adalah kebutuhan yang harus dimiliki oleh sistem atau proses apa saja yang akan dilakukan oleh sistem. Berikut kebutuhan fungsional pada sistem pemesanan menu di Toko Mansure.

- 1) Pelanggan dapat mengakses halaman *home*.
- 2) Pelanggan dapat mengakses halaman tagihan pembayaran.
- 3) Pelanggan dapat mengakses daftar menu.

- 4) Pelanggan dapat memesan menu.
- 5) Pelanggan dapat mengedit pesanan.
- 6) Pelanggan dapat menghapus pesanan.
- 7) Pelanggan dapat memberikan keterangan pada pesanan.
- 8) Pelanggan dapat melihat pesanan.
- 9) Koki dapat *login* sistem.
- 10) Koki dapat melihat pesanan pelanggan.
- 11) Koki dapat mengedit status pesanan jika sudah selesai dikerjakan.
- 12) Kasir dapat *login* sistem.
- 13) Kasir dapat mengelola transaksi pembayaran.
- 14) Kasir dapat mencetak setruk pembayaran.
- 15) Kasir dapat melihat laporan transaksi pembayaran.
- 16) Kasir dapat mencetak laporan transaksi pembayaran.
- 17) Super admin dapat *login* sistem.
- 18) Super admin dapat melihat halaman daftar menu.
- 19) Super admin dapat menambahkan menu baru.
- 20) Super admin dapat mengedit daftar menu.
- 21) Super admin dapat menghapus menu.
- 22) Super admin dapat melihat halaman data *user*.
- 23) Super admin dapat menambahkan *user* baru.
- 24) Super admin dapat *reset password user*.
- 25) Super admin dapat menghapus akun *user*.
- 26) Super admin dapat melihat pesanan.
- 27) Super admin dapat mengedit status pesanan.
- 28) Super admin dapat mengelola transaksi pembayaran.
- 29) Super admin dapat mencetak setruk pembayaran.
- 30) Super admin dapat melihat laporan transaksi pembayaran.
- 31) Super admin dapat mencetak laporan transaksi pembayaran.

B. Kebutuhan Non Fungsional

Kebutuhan Non fungsional adalah kebutuhan operasional. Kebutuhan Non fungsional pada sistem pemesanan menu di Toko Mansure adalah sebagai berikut.

1) Kebutuhan Keamanan

Sistem yang dibuat memiliki level pengguna yang berbeda. Karyawan akan diminta *login* menggunakan *username* dan *password* untuk mengakses bagiannya.

2) Kebutuhan Perangkat Keras

Kebutuhan perangkat keras dalam membangun sistem pemesanan makanan adalah sebagai berikut.

System Manufacturer : Lenovo

System Model : 20CLA200ID

Processor : Intel® Core™ i5-5200U

Memory : 8192MB RAM

3) Kebutuhan Perangkat Lunak

a. Sistem Operasi Windows 10

Sistem operasi windows 10 adalah sistem operasi pada laptop yang digunakan untuk membangun sistem pemesanan menu ini dan pembuatan laporan.

b. Visual Studio Code

Visual studio code adalah *code editor* yang berfungsi untuk menuliskan *source code* dalam pembuatan sistem pemesanan menu.

c. XAMPP versi 7.4.30

XAMPP adalah *software* web server lokal yang digunakan untuk membuat sebuah *website*.

d. Web browser

Web browser adalah *software* yang digunakan untuk mengakses halaman *website*. Web browser digunakan untuk menjalankan sistem yang telah dikembangkan yaitu sistem pemesanan menu karena sistem pemesanan menu ini berbasis *website*.

e. Star UML

Star UML adalah *software* yang digunakan untuk membuat pemodelan UML. Pada penelitian ini, star UML digunakan untuk membuat *use case diagram*.

f. Lucidchart

Lucidchart adalah *software* yang digunakan untuk membuat diagram. Pada penelitian ini lucidchart digunakan untuk membuat ERD.

g. Microsoft Office

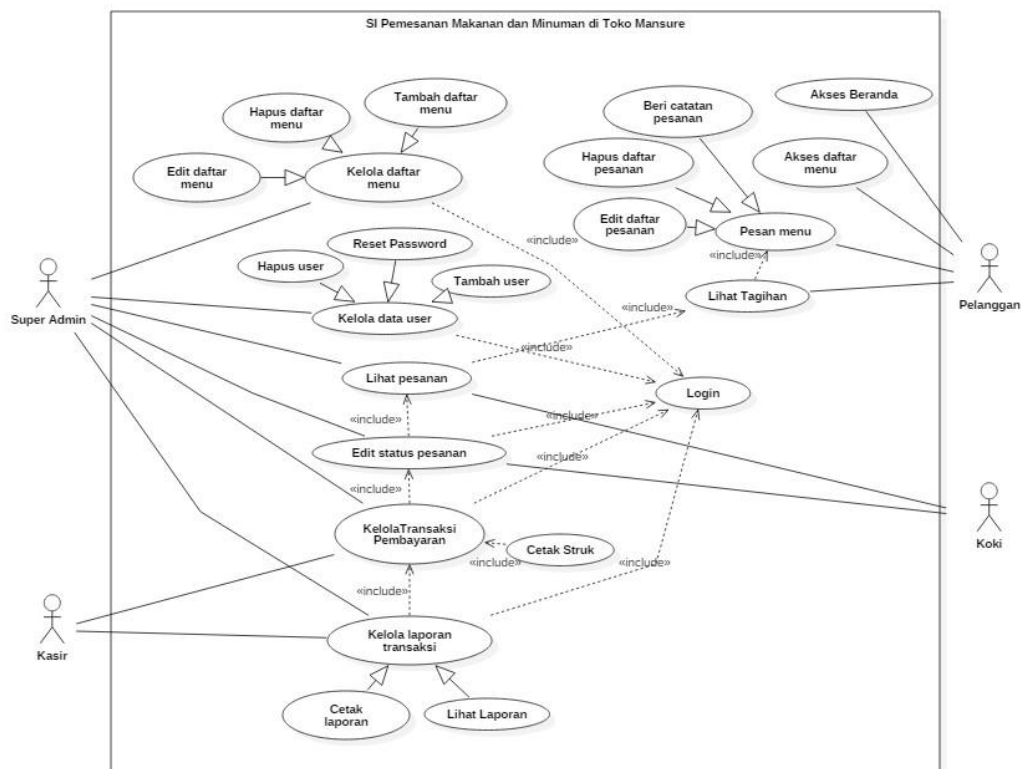
Microsoft office adalah *software* yang digunakan untuk mengolah kata menjadi dokumen. Pada penelitian ini, Microsoft office digunakan untuk membuat laporan.

2. Desain

Hasil yang didapatkan pada tahapan desain adalah desain *user interface*, diagram ERD, dan *use case*.

A. *Use Case*

Setelah menentukan analisis kebutuhan fungsional sistem maka fitur yang sistem divisualisasikan ke dalam *use case diagram*. Dengan adanya *use case diagram* memudahkan penggambaran proses yang dapat dilakukan oleh sistem. Diagram *use case* dapat dilihat pada Gambar 6.



Gambar 6. *Use Case Diagram*.

Pengguna dalam sistem pemesanan menu adalah pelanggan, koki, kasir, dan super admin. Setiap pengguna memiliki aktivitasnya masing-masing.

1) Pelanggan

Pelanggan dapat melihat halaman utama yang berisi daftar menu yang rekomendasi, melihat daftar menu, memesan menu, melihat pesanan dan dapat melihat tagihan pembayaran. Pada proses pemesanan menu pelanggan dapat mengedit, memberikan keterangan pemesanan, dan dapat menghapus pesanan.

2) Koki

Koki dapat *login* terlebih dahulu sebelum melihat daftar pesanan dan mengubah status pesanan. Status pesanan menunggu berubah menjadi selesai ketika koki telah selesai membuat pesanan tersebut.

3) Kasir

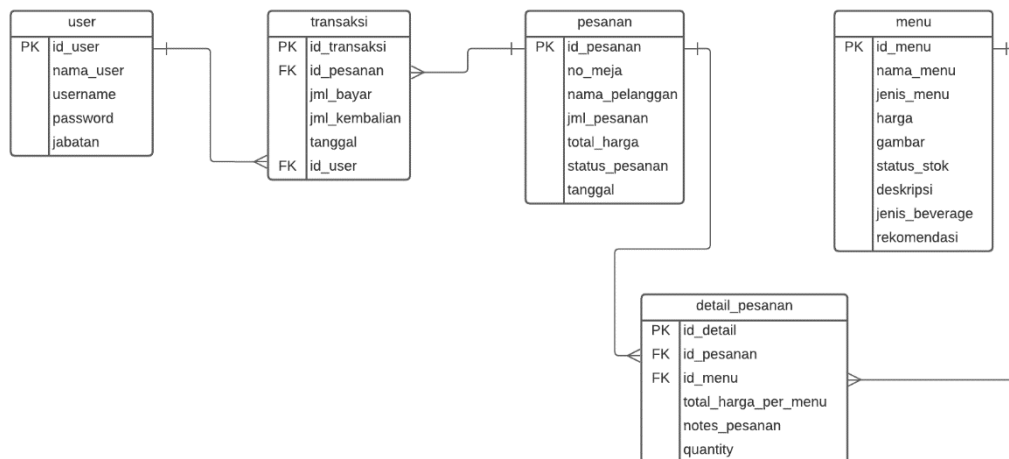
Kasir dapat *login* terlebih dahulu sebelum mengelola transaksi pembayaran dan mengelola laporan transaksi. Pada transaksi pembayaran kasir dapat mencetak setruk pembelian dan pada kelola laporan transaksi pembayaran kasir dapat melihat laporan dan mencetak laporan.

4) Super Admin

Super admin dapat *login* terlebih dahulu sebelum melakukan aktivitas di sistem. Super admin dapat melakukan seluruh kegiatan yang dapat dilakukan oleh kasir dan koki. Selain dapat melakukan tugas koki dan kasir, super admin dapat mengelola daftar menu dan akun karyawan.

B. ERD

Desain *entity relationship diagram* atau ERD dapat membantu menjelaskan bagaimana data-data yang ada pada sistem dapat dikelompokkan pada suatu entitas *database* dengan relasinya. Data yang ada pada sistem pemesanan di Toko Mansure dapat dilihat pada Gambar 7.



Gambar 7. Entity Relation Diagram.

C. Kamus Data

Kamus data digunakan untuk menjelaskan seluruh entitas pada *entity relationship diagram* atau ERD. Dengan adanya kamus data seluruh atribut yang ada di setiap entitas dapat menjelaskan maksudnya. Kamus data pada sistem ini adalah sebagai berikut.

1) Kamus Data Entitas *User*

Kamus data entitas *user* menjelaskan setiap atribut yang ada di entitas *user*. Kamus data entitas *user* dapat dilihat pada Tabel 2.

Tabel 2. Kamus Data Entitas *User*

Field	Type Data	Ukuran	Keterangan
Id	integer	5	Primary key tabel <i>user</i>
Nama_user	Varchar	20	Nama karyawan
Username	Varchar	10	Username untuk login karyawan
Password	Varchar	10	Password untuk login karyawan
Jabatan	Varchar	5	Kode karyawan untuk perannya dalam sistem

2) Kamus Data Entitas Menu

Kamus data entitas menu menjelaskan setiap atribut yang ada di entitas menu. Kamus data entitas menu dapat dilihat pada tabel struktur data pada Tabel 3.

Tabel 3. Kamus Data Entitas Menu

Field	Type Data	Ukuran	Keterangan
Id_menu	Integer	5	Primary key tabel menu
Nama_menu	Varchar	20	Nama menu

Tabel 3. (lanjutan)

Field	Tipe Data	Ukuran	Keterangan
Status_stok	Varchar	8	Status kesediaan menu kosong atau tersedia
Deskripsi	Varchar	50	Detail isi menu
Rekomendasi	Varchar	1	Menu yang direkomendasikan
Jenis_beverages	varchar	3	Jenis minuman
Harga	Integer	11	Harga menu
Jenis_menu	Varchar	20	Jenis menu
Gambar	Varchar	100	Gambar menu

3) Kamus Data Entitas Pesanan

Kamus data entitas pesanan menjelaskan setiap atribut yang ada di entitas pesanan. Kamus data entitas pesanan dapat dilihat pada tabel struktur data pada Tabel 4.

Tabel 4. Kamus Data Entitas Pesanan

Field	Tipe Data	Ukuran	Keterangan
Id_pesanan	Integer	5	<i>Primary key</i> tabel pesanan
No_meja	Integer	2	Nomor pada setiap meja
Nama_pelanggan	Varchar	15	Nama pelanggan
Jml_pesanan	Integer	5	Jumlah pesanan per menu
Total_harga_seluruh	Integer	11	Total harga keseluruhan yang harus dibayarkan oleh pelanggan
Status_pesanan	Varchar	8	Status pesanan menunggu, selesai, atau bayar
Tanggal	Datetime		Tanggal pemesanan menu

4) Kamus Data Entitas Transaksi

Kamus data entitas transaksi menjelaskan setiap atribut yang ada di entitas transaksi. Kamus data entitas transaksi dapat dilihat pada tabel struktur data pada Tabel 5.

Tabel 5. Kamus Data Entitas Transaksi

Field	Tipe Data	Ukuran	Keterangan
Id_transaksi	Integer	5	<i>Primary key</i> tabel transaksi
Id_pesanan	Integer	5	<i>Foreign key</i> pesanan
Jml_bayar	Integer	11	Nominal yang diberikan oleh pelanggan
Jml_kembalian	Integer	11	Nominal yang dikembalikan ke pelanggan
Id_user	Integer	5	<i>Foreign key</i> tabel user
Tanggal	Datetime		Tanggal transaksi pembayaran

5) Kamus Data Entitas Detail_pesanan

Kamus data entitas detail_pesanan menjelaskan setiap atribut yang ada di entitas detail_pesanan. Kamus data entitas detail_pesanan dapat dilihat pada tabel struktur data pada tabel 6.

Tabel 6. Kamus Data Entitas Detail_pesanan

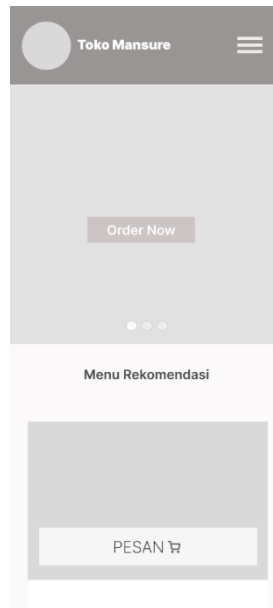
Field	Type Data	Ukuran	Keterangan
Id_detail	Integer	5	Primary key tabel detail_pesanan
Id_menu	Integer	5	Foreign key no_meja
Total_harga_per_menu	Integer	15	Jumlah harga per menu
Notes_pesanan	Varchar	50	Catatan pada pesanan per menu
Quantity	Integer	3	Jumlah pesanan per menu
Id_pesanan	Integer	5	Foreign key pesanan

D. User Interface

Desain rancangan desain *user interface* pada sistem ini adalah sebagai berikut:

a) Halaman *Home* Pelanggan

Pada halaman *home* pelanggan dapat melihat daftar menu yang direkomendasikan oleh Toko Mansure dan dapat masuk ke halaman daftar menu dengan menekan tombol *order now*. Jika pelanggan menekan tombol *order now* maka akan masuk ke halaman daftar menu dengan jenis menu *main course*. Pelanggan dapat mengakses fitur lainnya dengan cara menekan tombol yang bergaris tiga di pojok sebelah kanan atas. Terdapat fitur untuk melihat jenis menu lainnya dan keranjang pesanan yang dapat diakses oleh pelanggan. Gambar rancangan halaman *home* dapat dilihat pada Gambar 8.



Gambar 8 Halaman *Home*.

b) Halaman Daftar Menu Pelanggan

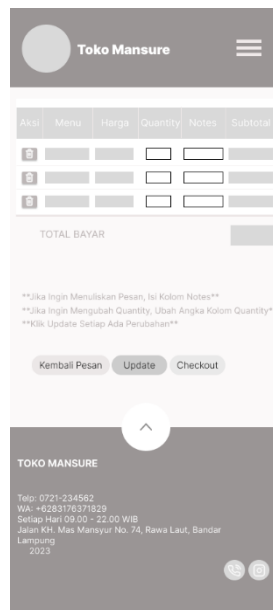
Pada halaman daftar menu terdapat berbagai macam menu sesuai dengan jenisnya. Pelanggan dapat memilih jenis menu yang ingin dilihat. Jika pelanggan ingin memesan menu tersebut, pelanggan dapat menekan tombol pesan maka, pesanan akan tersimpan. Gambar rancangan halaman daftar menu pelanggan dapat dilihat pada Gambar 9.



Gambar 9. Halaman Daftar Menu Pelanggan.

c) Halaman Keranjang Pesanan

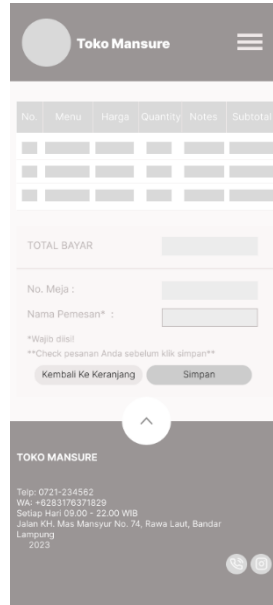
Pada halaman keranjang terdapat daftar pesanan pelanggan yang berisi nama menu, jumlah pesanan, catatan pesanan, harga, dan aksi untuk menghapus pesanan. Pelanggan dapat mengubah *quantity* dan catatan pesanan. Terdapat tombol *update* untuk *update* setiap perubahan yang ada, tombol kembali untuk kembali ke daftar menu, dan tombol *checkout* untuk beralih ke halaman *checkout*. Gambar rancangan halaman keranjang dapat dilihat pada Gambar 10.



Gambar 10. Halaman Keranjang Pesanan.

d) Halaman *Checkout*

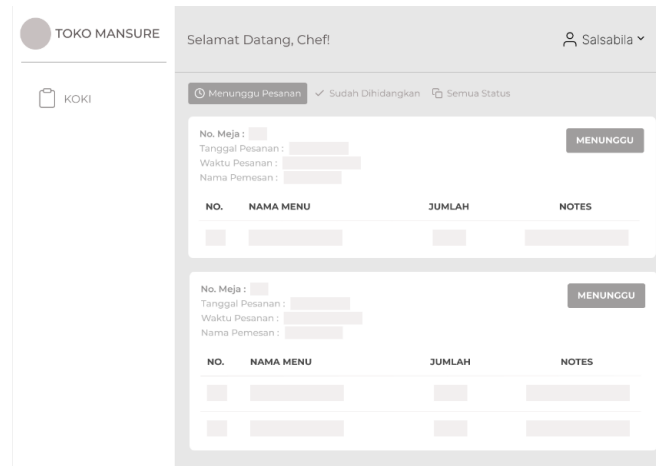
Pada halaman *checkout* terdapat daftar menu yang akan dipesan dan *form* untuk mengisi nama pemesan. Nomor meja di dapatkan ketika pelanggan *scan barcode* yang ada di meja pelanggan. Sehingga pelanggan tidak perlu mengisi nomor meja tersebut. Rancangan halaman *checkout* dapat dilihat pada Gambar 11.



Gambar 11. Halaman *Checkout*.

e) Halaman Daftar Pesanan

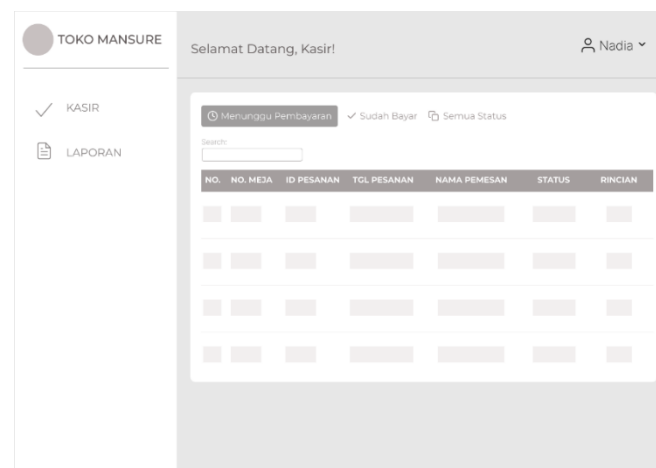
Pada halaman ini koki dapat melihat seluruh daftar pesanan dari pelanggan dengan status menunggu dan selesai. Status menunggu adalah pesanan yang baru masuk dan status selesai adalah pesanan yang sudah dibuat oleh koki yang akan dihidangkan ke pelanggan. Koki dapat melihat seluruh daftar pesanan dengan status selesai dengan menekan tulisan sudah dihidangkan dan untuk melihat semua status dengan menekan tulisan semua status. Koki dapat melihat detail pesanan dan jika pesanan sudah selesai dibuat maka koki dapat menekan tombol tulisan “menunggu” untuk mengubah status pesanan menjadi selesai. Gambar untuk rancangan halaman daftar pesanan dapat dilihat pada Gambar 12.



Gambar 12. Halaman Daftar Pesanan.

f) Halaman Daftar Pembayaran

Pada halaman ini kasir dapat melihat seluruh daftar pembayaran dengan status selesai dan bayar. Status selesai adalah pesanan yang telah dihidangkan tetapi belum dibayar oleh pelanggan. Status bayar adalah pesanan yang sudah dibayarkan oleh pelanggan. Kasir dapat mencari nama pelanggan, id pesanan, ataupun data lainnya yang diinginkan yang ada di tabel pesanan. Jika kasir ingin melihat seluruh daftar pembayaran dengan status bayar maka, kasir dapat menekan tulisan sudah bayar dan jika ingin seluruh status dapat menekan tulisan semua status. Gambar untuk rancangan halaman daftar pembayaran dapat dilihat pada Gambar 13.



Gambar 13. Halaman Daftar Pembayaran.

g) Halaman Transaksi Pembayaran

Pada halaman ini kasir dapat melihat detail pesanan pelanggan dan melakukan proses transaksi dengan menginputkan total yang dibayarkan oleh pelanggan. Rancangan desain halaman transaksi pembayaran dapat dilihat pada Gambar 14.

Gambar 14. Halaman Transaksi Pembayaran.

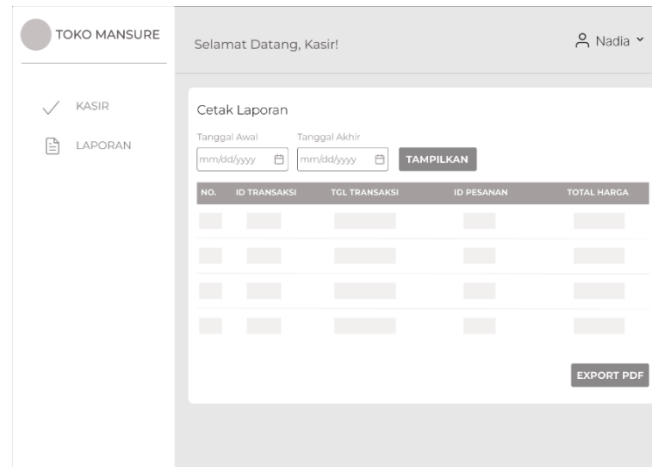
h) Halaman Setruk

Kasir dapat mencetak setruk jika pelanggan sudah melakukan transaksi pembayaran. Pada setruk terdapat informasi mengenai daftar pesanan pelanggan, total yang harus dibayarkan, total yang diberikan pelanggan, kembalian, nama kasir yang sedang bertugas, nama pelanggan, nomor meja, dan nomor pesanan pelanggan. Rancangan halaman setruk dapat dilihat pada Gambar 15.

Gambar 15. Halaman Setruk.

i) Halaman Daftar Laporan

Pada halaman ini kasir dapat melihat daftar transaksi pelanggan secara keseluruhan maupun berdasarkan tanggal yang diinginkan dan dapat mencetak laporan tersebut. Gambar untuk rancangan halaman daftar laporan dapat dilihat pada Gambar 16.



Gambar 16. Halaman Daftar Laporan.

j) Halaman Cetak Laporan

Laporan dapat dicetak secara keseluruhan atau berdasarkan tanggal yang diinginkan dalam bentuk PDF. Isi pada halaman laporan tersebut adalah id transaksi, id pesanan, tanggal transaksi, total pembayaran per id transaksi, dan jumlah dana yang didapatkan oleh Toko Mansure. Selain itu, terdapat tanggal periode jika ingin mencetak berdasarkan tanggal atau periode tertentu. Gambar rancangan halaman cetak laporan dapat dilihat pada Gambar 17.

TOKO MANSURE
Jalan KH. Mas Mansur No. 74, Rw Laut, Enggal, Kota Bandar Lampung, Lampung

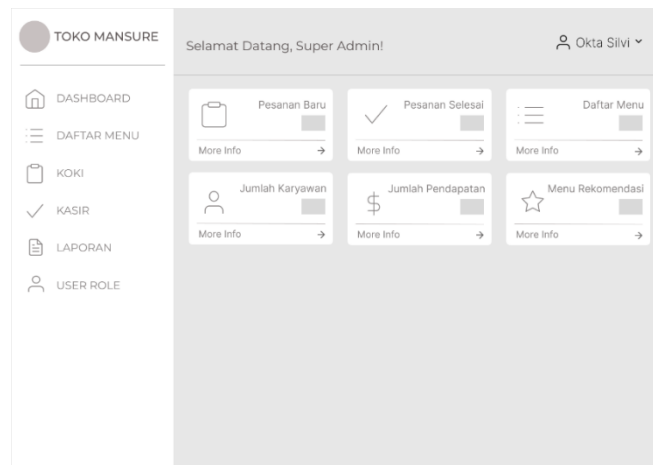
Laporan Transaksi Keuangan
Periode s/d

No.	ID Transaksi	Tanggal Transaksi	ID Pesanan	Total Pembayaran

Gambar 17. Halaman Cetak Laporan.

k) Halaman *Dashboard*

Pada halaman ini super admin dapat melihat informasi mengenai total pesanan yang menunggu, total pesanan yang telah dihidangkan atau pesanan yang belum bayar, total daftar menu yang tersedia di Toko Mansure, total karyawan jumlah pendapatan, dan total menu yang direkomendasikan. Gambar rancangan halaman *dashboard* dapat dilihat pada Gambar 18.

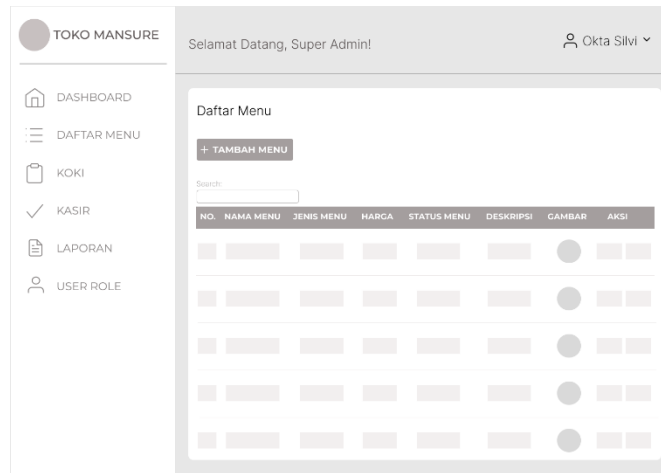


Gambar 18. Halaman *Dashboard*.

l) Halaman Daftar Menu

Pada halaman ini super admin dapat melihat daftar menu yang ada di Toko Mansure dan dapat mencari data berdasarkan nama menu, jenis menu, dan data lainnya yang ada di daftar tabel. Terdapat pula beberapa tombol untuk

menambahkan, mengedit, dan menghapus menu. Gambar rancangan halaman daftar menu dapat dilihat pada Gambar 19.



Gambar 19. Halaman Daftar Menu

m) Halaman Tambah Menu

Pada halaman tambah menu terdapat *form* menu yang dapat diinputkan sesuai dengan menu yang akan ditambahkan. *Form* tambah menu tersebut berisi deskripsi, gambar, status kesediaan, harga, jenis menu, status rekomendasi, dan nama menu. tombol aksi untuk edit, hapus, dan tambah menu. Gambar rancangan halaman tambah menu dapat dilihat pada Gambar 20.

Gambar 20. Halaman Tambah Menu.

n) Halaman Edit Menu

Pada halaman ini super admin dapat mengedit menu yang ada. Rancangan halaman edit menu dapat dilihat pada Gambar 21.

Gambar 21. Halaman Edit Menu.

o) Hapus Menu

Pada bagian ini super admin dapat menghapus menu yang sudah tidak digunakan lagi. Gambar rancangan jika akan menghapus menu dapat dilihat pada Gambar 22.

Gambar 22. Hapus Menu.

p) Halaman Tambah Akun dan Daftar Karyawan

Pada halaman ini super admin dapat melihat daftar akun karyawan yang ada di Toko Mansure dan dapat menambahkan akun. Setiap karyawan memiliki satu

akun untuk mengakses sistem agar dapat mengerjakan tugasnya. Gambar rancangan halaman tambah akun dan daftar karyawan dapat dilihat pada Gambar 23.

Gambar 23. Halaman Tambah Akun dan Daftar Karyawan.

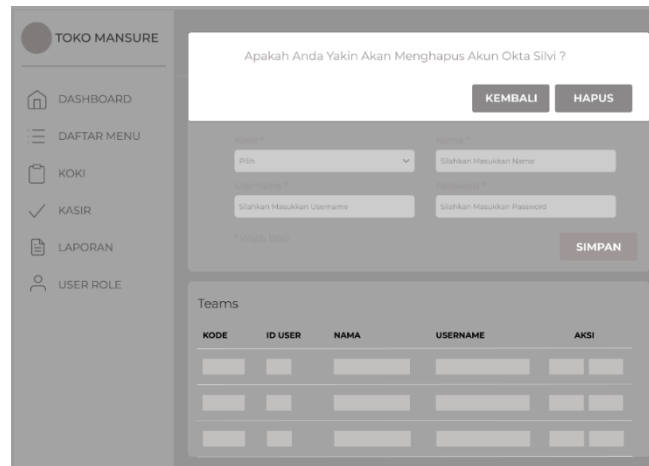
q) *Reset Password*

Pada bagian ini super admin dapat *reset password* dan akan diminta untuk memasukkan *password* baru. Gambar rancangan jika akan *reset password* dapat dilihat pada Gambar 24.

Gambar 24. *Reset Password*.

r) *Hapus Akun Karyawan*

Pada bagian ini super admin dapat menghapus akun karyawan yang sudah tidak bekerja di Toko Mansure lagi. Gambar rancangan jika akan hapus akun dapat dilihat pada Gambar 25.



Gambar 25. Hapus Akun Karyawan.

s) Halaman *Login* karyawan

Pada halaman *login* karyawan terdapat *form* input *username* dan *password* untuk masuk ke sistem. Gambar rancangan halaman *login* karyawan dapat dilihat pada Gambar 26.



Gambar 26. Halaman *Login* Karyawan.

3. Pengkodean

Pembuatan sistem pemesanan menu di Toko Mansure menggunakan *framework* CodeIgniter versi 4 dengan menggunakan Bahasa pemrograman PHP. Pengkodean pada penelitian dibagi menjadi dua bagian yaitu *back-end* dan *front-end*. Pada bagian *front-end* dikembangkan oleh Okta Silvi. *Back-end* digunakan

untuk memfungsikan tampilan yang telah dibuat oleh Okta Silvi sehingga seluruh fitur yang ada pada sistem dapat digunakan sebagaimana mestinya.

4. Pengujian

Tahap akhir dari metode *extreme programming* adalah pengujian. Pengujian yang dilakukan adalah untuk menguji setiap kriteria pada *back-end* yaitu dari sisi performa, integrasi, dan keamanan. Pengujian performa menggunakan alat Apache Jmeter, pengujian integrasi menggunakan alat selenium IDE, dan pengujian keamanan menggunakan alat OWASP ZAP.

a) Pengujian performa

Pengujian performa dibagi menjadi dua *role* yaitu pelanggan dan karyawan. Pada pengujian performa akan dicari berapa banyak *user* sistem tanpa adanya penurunan performa. Sehingga dapat diketahui pada *user* keberapa sistem akan mengalami penurunan performa. Tahapan yang pertama pada *role* pelanggan adalah dengan menambahkan utas dengan tiga komponen yaitu *threads*, *ramp-up periode*, dan *looping count*. Tahapan kedua adalah dengan menambahkan *sampler* dengan *HTTP request*. *Listener* yang ditambahkan pada pengujian ini adalah *summary report*. *Test plan* pada *HTTP request* untuk bagian pelanggan dapat dilihat pada Tabel 7.

Tabel 7. *Test Plan HTTP Request* Pelanggan

Nama	Path
<i>HomePelanggan</i>	<i>index.php/</i>
<i>MenuPelanggan</i>	<i>menu</i>
<i>KeranjangPelanggan</i>	<i>keranjang-nih</i>
<i>CheckoutPelanggan</i>	<i>cart/tagihan</i>
<i>Pasta</i>	<i>menu/pasta</i>
<i>MainCourse</i>	<i>menu/maincourse</i>
<i>Ucaan</i>	<i>menu/uncaanmansure</i>
<i>Snacks</i>	<i>menu/snacks</i>
<i>Beverages</i>	<i>menu/beverages</i>
<i>SweetTooth</i>	<i>menu/sweetooth</i>

Tahapan yang pertama pada *role* karyawan adalah dengan menambahkan utas dengan tiga komponen yaitu *threads*, *ramp-up periode*, dan *looping count*. Tahapan kedua adalah dengan menambahkan *sampler* dengan *HTTP request*. *Listener* yang ditambahkan pada pengujian ini adalah *summary report*. *Test plan* pada *HTTP request* untuk bagian pelanggan dapat dilihat pada Tabel 8.

Tabel 8. *Test Plan HTTP Request* Karyawan

Nama	Path
<i>Login</i>	halaman- <i>login</i> -pelayan
<i>DashboardAdmin</i>	<i>dashboard</i>
<i>DaftarMenu</i>	<i>dashboard/home</i> -kasir
<i>TambahMenu</i>	<i>dashboard/tambah</i> -menu
<i>EditMenu</i>	<i>dashboard/edit</i> -menu/171
<i>Laporan</i>	<i>dashboard/laporan</i>
<i>DetailLaporan</i>	<i>index.php/dashboard/cetak</i>
<i>ExportPDF</i>	<i>dashboard/export</i> -pdf
<i>UserRole</i>	<i>dashboard/tambah</i> -user
<i>Kasir</i>	<i>dashboard/transaksi</i>
<i>Koki</i>	<i>dashboard/koki</i>
<i>TransaksiKasir</i>	<i>dashboard/pembayaran</i> /1431
<i>Setruk</i>	<i>index.php/dashboard/setruk</i> /1430

b) Pengujian integrasi

Pengujian integrasi dilakukan untuk melihat apakah *back-end* pada sistem sudah terintegrasi dengan baik terhadap sistem lainnya seperti *database*, *front-end*, dan sebagainya sebelum digunakan oleh *user*. Tahapan pertama yang harus dilakukan sebelum pengujian integrasi adalah membuat rencana pengujian yaitu menentukan akan menggunakan *tools* apa. *Tools* yang digunakan untuk melakukan pengujian adalah selenium ide. Tahapan kedua adalah membuat skenario uji kemudian dilakukan pengujian. Daftar skenario uji dapat dilihat pada Tabel 9.

Tabel 9. Skenario Uji Integrasi

Test Case	Skenario Uji
<i>Home</i>	a. Membuka halaman <i>home</i> b. Menekan tombol <i>home</i>
Lihat menu pelanggan	a. Membuka halaman sistem b. Menekan tombol order now c. Menekan tombol untuk memilih jenis menu
Tambah menu pelanggan	a. Membuka halaman sistem b. Menekan tombol pesan

Tabel 9. (lanjutan)

<i>Test Case</i>	Skenario Uji
Edit menu pelanggan	<ul style="list-style-type: none"> a. Membuka halaman sistem b. Menekan tombol keranjang c. Memasukkan catatan pesanan d. Mengubah <i>quantity</i> pesanan e. Menekan tombol <i>update</i>
Hapus menu pelanggan	<ul style="list-style-type: none"> a. Membuka halaman sistem b. Menekan tombol keranjang c. Menekan tombol dengan <i>icon</i> tempat sampah
Pesan menu	<ul style="list-style-type: none"> a. Membuka halaman sistem b. Menekan tombol keranjang c. Menekan tombol <i>checkout</i> d. Memasukkan nama pemesan
<i>Login</i> admin	<ul style="list-style-type: none"> a. Membuka halaman <i>login</i> b. Memasukkan <i>username</i> c. Memasukkan <i>password</i>
<i>Dashboard</i> admin	<ul style="list-style-type: none"> a. Membuka halaman <i>login</i> b. Memasukkan <i>username</i> c. Memasukkan <i>password</i> d. Menekan tombol <i>dashboard</i>
Daftar menu admin	<ul style="list-style-type: none"> a. Membuka halaman <i>login</i> b. Memasukkan <i>username</i> c. Memasukkan <i>password</i> d. Menekan tombol daftar menu e. Memasukkan nama pencarian f. Memilih <i>pagination</i>
Tambah menu	<ul style="list-style-type: none"> a. Membuka halaman <i>login</i> b. Memasukkan <i>username</i> c. Memasukkan <i>password</i> d. Menekan tombol daftar menu e. Menekan tombol tambah menu f. Memasukkan informasi mengenai menu yang akan ditambah g. Menekan tombol simpan

Tabel 9. (lanjutan)

<i>Test Case</i>	Skenario Uji
Edit menu	<ul style="list-style-type: none"> a. Membuka halaman <i>login</i> b. Memasukkan <i>username</i> c. Memasukkan <i>password</i> d. Menekan tombol daftar menu e. Menekan tombol edit menu f. Mengubah informasi menu g. Menekan tombol simpan
Hapus menu	<ul style="list-style-type: none"> a. Membuka halaman <i>login</i> b. Memasukkan <i>username</i> c. Memasukkan <i>password</i> d. Menekan tombol daftar menu e. Menekan tombol dengan ikon hapus
Tambah <i>user</i>	<ul style="list-style-type: none"> a. Membuka halaman <i>login</i> b. Memasukkan <i>username</i> c. Memasukkan <i>password</i> d. Menekan tombol <i>user role</i> e. Memasukkan informasi mengenai <i>user</i> f. Menekan tombol simpan
Hapus <i>user</i>	<ul style="list-style-type: none"> a. Membuka halaman <i>login</i> b. Memasukkan <i>username</i> c. Memasukkan <i>password</i> d. Menekan tombol <i>user role</i> e. Menekan tombol dengan ikon hapus
<i>Reset password</i>	<ul style="list-style-type: none"> a. Membuka halaman <i>login</i> b. Memasukkan <i>username</i> c. Memasukkan <i>password</i> d. Menekan tombol <i>user role</i> e. Menekan tombol dengan <i>icon reset</i> f. Memasukkan <i>password</i> yang baru
Koki	<ul style="list-style-type: none"> a. Membuka halaman <i>login</i> b. Memasukkan <i>username</i> c. Memasukkan <i>password</i> d. Menekan tombol pada bagian rincian untuk mengubah status

Tabel 9. (lanjutan)

<i>Test Case</i>	Skenario Uji
Kasir	a. Membuka halaman <i>login</i> b. Memasukkan <i>username</i> c. Memasukkan <i>password</i> d. Memasukkan nama pencarian e. Memilih <i>pagination</i>
Transaksi kasir	a. Membuka halaman <i>login</i> b. Memasukkan <i>username</i> c. Memasukkan <i>password</i> d. Menekan tombol pada bagian rincian e. Memasukkan nominal yang dibayarkan oleh pelanggan f. Menekan tombol simpan
Setruk	a. Membuka halaman <i>login</i> b. Memasukkan <i>username</i> c. Memasukkan <i>password</i> d. Menekan tombol pada bagian rincian e. Memasukkan nominal yang dibayarkan oleh pelanggan f. Menekan tombol simpan g. Menekan tombol <i>print</i> h. Menekan tombol kembali
Daftar laporan	a. Membuka halaman <i>login</i> b. Memasukkan <i>username</i> c. Memasukkan <i>password</i> d. Menekan tombol laporan e. Memasukkan tanggal yang ingin dilihat f. Menekan tombol tampilkan
<i>Export</i> laporan	a. Membuka halaman <i>login</i> b. Memasukkan <i>username</i> c. Memasukkan <i>password</i> d. Menekan tombol laporan e. Memasukkan tombol <i>export</i> f. Menekan tombol <i>print</i>

c) Pengujian keamanan

Pengujian keamanan dengan OWASP ZAP cukup dengan memasukkan link URL agar ZAP dapat memindai kerentanan apa saja yang dapat terjadi pada sistem pemesanan di Toko Mansure.

3.2.3 Penulisan Laporan

Tujuan tahapan penulisan laporan adalah untuk menyampaikan informasi mengenai penelitian yang telah dilakukan dan berisi seluruh dokumentasi kegiatan. Dengan adanya laporan penelitian ini dapat memudahkan pembaca mengetahui kegiatan penelitian yang dilakukan dan dapat dijadikan sebagai bahan referensi untuk penelitian berikutnya.

V KESIMPULAN DAN SARAN

5.1 Kesimpulan

Kesimpulan yang didapatkan dari penelitian ini adalah sebagai berikut.

1. Telah dikembangkan *back-end* dari sistem informasi pemesanan menu berbasis *website* menggunakan *framework* CodeIgniter 4.
2. *Back-end* pada sistem ini telah mengintegrasikan tampilan *front-end* dan menjalankan fungsionalitas dari berbagai fitur yang ada pada setiap *role*. Fitur tersebut adalah pemesanan menu, mengelola daftar menu, mengelola data *user*, mengelola transaksi, mengelola pesanan, dan mengelola laporan yang dilakukan di Toko Mansure.
3. Keamanan pada sistem pemesanan yang ada di Toko Mansure memiliki 1 kerentanan dengan tingkat risiko yang tinggi, 5 dengan tingkat menengah, 3 dengan tingkat rendah, dan informatif sebanyak 4.
4. Sistem informasi pemesanan menu telah terintegrasi baik dengan sistem lainnya.
5. Sistem informasi pemesanan menu pada bagian pelanggan akan terjadi penurunan performa dengan *user* lebih dari 244 *user* dan pada bagian karyawan jika lebih dari 216 *user*.
6. Sistem informasi pemesanan menu ini hanya dapat digunakan oleh pelanggan Toko Mansure saja.

5.2 Saran

Berdasarkan hasil yang didapatkan dari implementasi sistem pemesanan menu ini maka didapatkan saran untuk pengembangan selanjutnya yaitu.

1. Bisa dilakukan pemesanan dari rumah atau jarak jauh dan diantar sampai ke tempat tujuan.
2. Saat *reset password* akun karyawan menggunakan *two factor authentication* agar tidak sembarang orang mudah menggantinya.
3. Menambahkan estimasi waktu pesanan akan datang.
4. Meningkatkan atau memperbaiki keamanan pada sistem karena pada saat pengujian terdapat kerentanan untuk disusupi.

DAFTAR PUSTAKA

- Aliya, H. (2021, Februari 04). *Bidang Profesi: Software Engineering: Tech & Data: System Integration Testing, Metode Pengujian yang Akan Mempermudah Pekerjaanmu*. Retrieved from glints: <https://glints.com/id/lowongan/system-integration-testing-sit-adalah/>
- Amalia, A., Hamidah, S. W., & Kristanto, T. (2021). Pengujian Black Box Menggunakan Teknik Equivalence Partitions Pada Aplikasi E-Learning Berbasis Web. *Building of Informatics, Technology and Science (BITS) Volume 3, No 3*, 269-270.
- ANDARU, A. (2018). PENGERTIAN DATABASE SECARA UMUM. *Fakultas Komputer Section Class Content*, 1-4.
- Arhandi, P. P. (2016). PENGEMBANGAN SISTEM INFORMASI PERIJINAN TENAGA KESEHATAN DENGAN MENGGUNAKAN METODE BACK END DAN FRONT END. *Jurnal Teknologi Informasi*, 2.
- Ariyanti, L., Satria, M. N., & Alita, D. (2020). SISTEM INFORMASI AKADEMIK DAN ADMINISTRASI DENGAN METODE EXTREME PROGRAMMING PADA LEMBAGA KURSUS DAN PELATIHAN. *Jurnal Teknologi dan Sistem Informasi (JTSI)*, 2-3.
- Arviana, G. N. (2020, Desember 23). *Bidang Profesi: Product: Product Development*. Retrieved from glints: <https://glints.com/id/lowongan/unit-testing-adalah/>

- Asaniyah, N. (2017). PELESTARIAN INFORMASI KOLEKSI LANGKA: Digitalisasi, Restorasi, Fumigasi. 5.
- Astuti, R. (2009). Pemodelan Analisis Berorientasi Objek Dengan Use Case. *Media Informatika*, 3-7.
- Azura. (2023, Juni 14). Retrieved from Azura Labs: <https://azuralabs.id/programing/panduan-lengkap-tentang-backend-testing>
- Budiarti, Y. (2020). IMPLEMENTASI METODE EXTREME PROGRAMMING UNTUK MERANCANG SISTEM INFORMASI PENDAFTARAN SISWA BARU BERBASIS WEB PADA SMK MULTIMEDIA MANDIRI JAKARTA. *Jurnal Ilmiah Fakultas Sains dan Teknologi, Universitas Labuhanbatu*, 2.
- Dendi, M., & Nurjaya. (2022). Sistem Informasi Pemesanan Menu Makanan Pada Restoran Eighteen Pies Berbasis Web . *Jurnal Ilmu Komputer dan Sains* .
- Dewi, L. P., Indahyanti, U., & S, Y. H. (2012). PEMODELAN PROSES BISNIS MENGGUNAKAN ACTIVITY DIAGRAM UML DAN BPMN (STUDI KASUS FRS ONLINE). 2-4.
- Erdani, B., Aditia, F. D., Rodiah, S., Ciptyasih, & Santi, I. H. (2019). Sistem Aplikasi Kamus Istilah Bahasa Pemrograman PHP Menggunakan Algoritma Brute Force. *Jurnal Multimedia & Artificial Intelligence*, 2.
- Fahrullah. (2021). IMPLEMENTASI PENGUJIAN BLACK BOX PADA SISTEM INFORMASI MONITORING AKADEMIK DENGAN PENDEKATAN TEKNIK EQUIVALENCE PARTITIONS. *Jurnal Teknosains Kodepena* , 1-5.
- Febrian, V., Ramadhan, M. R., Faisal, M., & Saifudin, A. (2020). Pengujian pada Aplikasi Penggajian Pegawai dengan menggunakan Metode Blackbox. *Jurnal Informatika Universitas Pamulang*, 2.
- Gill, N. S. (2022, November 17). Retrieved from xenonstack: <https://www.xenonstack.com/insights/what-is-system-testing>

- Hamilton, T. (2023, Mei 6). Retrieved from <https://www.guru99.com/integration-testing.html>
- Haryanto, D., & Argadila, D. (2019). SISTEM INFORMASI PENGARSIPAN DATA KONSUMEN DI PT. DINASTI PERTIWI “PERUMAHAN DEWASARI”. *Jurnal Teknik Informatika*, 7.
- Hendini, A. (2016). PEMODELAN UML SISTEM INFORMASI MONITORING PENJUALAN DAN STOK BARANG (STUDI KASUS: DISTRO ZHEZHA PONTIANAK) . *JURNAL KHATULISTIWA INFORMATIKA*, 2.
- Hristovski, D., & Kotevski, G. (2021, Desember 6). *IWConnect : Blog : Pengantar Pengujian Keamanan dengan OWASP ZAP*. Retrieved from IWConnect: <https://iwconnect.com/introduction-to-security-testing-with-owasp-zap/>
- Juliandiny, S. (2023, Maret 8). *Blog: System Integration Testing: Definisi, Tujuan, Dan Manfaatnya*. Retrieved from Pacmann: <https://pacmann.io/blog/system-integration-testing>
- Khader, R. A. (2022, Oktober 28). Retrieved from codingninja/studios: <https://www.codingninjas.com/codestudio/library/what-is-jmeter-thread-group>
- Khan, M. E. (2011). DIFFERENT APPROACHES TO BLACK BOX TESTING TECHNIQUE FOR FINDING ERRORS. *Software Engineering & Applications Vol.2, No.4*, 31.
- Kurniawan, I., Humaira, & Rozi, F. (2020). REST API Menggunakan Node JS pada Aplikasi Transaksi Jasa Elektronik Berbasis Android. *Jurnal Ilmiah Teknologi Sistem Informasi*, 2.
- Kushchi, S. (2022, September 2022). Retrieved from Jit: <https://www.jit.io/blog/6-essential-steps-to-use-owasp-zap-for-penetration-testing>
- Kusuma, A. P., & Prasetya, K. A. (2017). PERANCANGAN DAN IMPLEMENTASI E-COMMERCEUNTUK PENJUALAN BAJU ONLINE BERBASIS ANDROID. *Jurnal Antivirus, Vol. 11No. 1*, 2.

- Lawrence, A. (2021, Oktober 17). *Home: Web Development & Desain: Backend Developer: 5+ Skill yang Harus Dikuasai untuk Jadi Profesional*. Retrieved from Niagahoster Blog: <https://www.niagahoster.co.id/blog/backend-developer/>
- Lumbantobing, H. (2021, Agustus 20). Retrieved from medium: <https://hariatylumbantobing.medium.com/analisis-hasil-performance-test-pada-jmeter-1d7a432906d7>
- Ma'rifat, I. S., & Kesuma, C. (2018). PENGEMBANGAN SISTEM PAKAR MENDETEKSI PENYAKIT PENCERNAAN MENGGUNAKAN METODE NAIVE BAYES BERBASIS WEB. *Jurnal Evolusi*, 2.
- Mohanan, R. (2022, November 21). Retrieved from spiceworks: <https://www.spiceworks.com/tech/devops/articles/what-is-integration-testing/>
- Monaselidze, G. (2022, Mei 13). Retrieved from mailtrap: <https://mailtrap.io/blog/back-end-testing/>
- Mustaqbal, M. S., Firdaus, R. F., & Rahmadi, H. (2015). PENGUJIAN APLIKASI MENGGUNAKAN BLACK BOX TESTING BOUNDARY VALUE ANALYSIS (Studi Kasus : Aplikasi Prediksi Kelulusan SNMPTN). *Jurnal Ilmiah Teknologi Informasi Terapan*, 2-4.
- Nader, K. (2023, Juni 12). *Rumah: Blog: Berbagai Jenis Pengujian Perangkat Lunak*. Retrieved from AppMaster: <https://appmaster.io/id/blog/jenis-pengujian-perangkat-lunak>
- Nader, Y. (2023, April 17). *Home: Articles: Information Security & Ethical Hacking / top-10-open-source-security-testing-tools-for-web-applications*. Retrieved from hackr.io: <https://hackr.io/blog/top-10-open-source-security-testing-tools-for-web-applications>
- Nanda, V. (2021, September 22). Retrieved from tutorialspoint: <https://www.tutorialspoint.com/backend-testing-tutorial-what-is-tools-examples>

- Novitasari, Y. S., Adrian, Q. J., & Kurnia, W. (2021). RANCANG BANGUN SISTEM INFORMASI MEDIA PEMBELAJARAN BERBASIS WEBSITE (STUDI KASUS: BIMBINGAN BELAJAR DEPOTLOOD). *Jurnal Teknologi dan Sistem Informasi*, 2.
- Rahmalia, N. (2020, November 18). *Career Advice: Career Hacks: Grey Box Testing, Uji Software yang Gabungkan Metode Black Box dan White Box*. Retrieved from glints: <https://glints.com/id/lowongan/grey-box-testing/>
- Rahmayu, M. (2016). Rancang Bangun Sistem Informasi Pada Rumah Sakit Dengan Layanan Intranet Menggunakan Metode Waterfall. *Jurnal Evolusi*, 2-3.
- Sallaby, A. F., & Kanedi, I. (2020). Perancangan Sistem Informasi Jadwal Dokter Menggunakan Framework Codeigniter. *Jurnal Media Infotama*, 1.
- Saputri, Z. R., Oktavia, A. N., Ramdhani, L. S., & Suherman, A. (2019). RANCANG BANGUN SISTEM INFORMASI PEMESANAN MAKANAN BERBASIS WEB PADA CAFE SURABIKU. *Jurnal Teknologi dan Informasi*.
- Setiawan, R. E. (2019, Maret Minggu, 24). *BLOG: INFO TEKNOLOGI: APA ITU LOAD TESTING DAN BAGAIMANA MELAKUKANNYA*. Retrieved from situstarget: <https://www.situstarget.com/blog/apa-itu-load-testing/>
- Setiyani, H. (2019). APLIKASI PEMESANAN MAKANAN BERBASIS WEB DI RAMEN SHINJU -RANJU-.
- Shethi, S. (2022, Oktober 12). Retrieved from Geekflare: <https://geekflare.com/apache-jmeter-guide/>
- Shofa, N. M. (2022, July 27). *kitalulus*. Retrieved from <https://www.kitalulus.com/seputar-kerja/back-end-developer-adalah>
- Sholeh, M., Aji, W. L., Riady, Y., & Qasthari, B. L. (2022). Pengelolaan Pemesanan Menu Makanan Menggunakan Framework Flask Python. *Jurnal Teknik Informatika dan Sistem Informasi* .

- Siregar, Y. B. (2019). DIGITALISASI ARSIP UNTUK EFISIENSI PENYIMPANAN DAN AKSESIBILITAS . *Jurnal Administrasi dan Kesekretarisan*, 6-7.
- Sofiani, I., & Nurhidayat, A. I. (2019). siRANCANG BANGUN APLIKASI E-MARKETPLACE HASIL PERTANIAN BERBASIS WEBSITE DENGAN MENGGUNAKAN FRAMEWORK CODEIGNITER. *Jurnal Manajemen Informatika*, 2-3.
- Sukmawati, R., & Priyadi, Y. (2019). Perancangan Proses Bisnis Menggunakan UML Berdasarkan Fit/Gap Analysis Pada Modul Inventory Odoo. *Jurnal Ilmiah Penelitian dan Penerapan Teknologi Sistem Informasi*, 3.
- Susanti, S., Junianto, E., & Rachman, R. (2017). Implementasi Framework Laravel Pada Aplikasi Pengolah Nilai Akademik Berbasis Web. *JURNAL INFORMATIKA*, 3.
- Unadkat, J. (2023, Febuari 12). *Panduan Beranda: Memulai Selenium IDE*. Retrieved from *BrowserStack*: <https://www.browserstack.com/guide/what-is-selenium-ide>
- Utomo, E. W., & Suryana, T. (2022). Sistem Informasi Pergudangan Dengan Menggunakan Teknologi RFID DI PT. BINTANG SIDORAYA . 2.
- Wahyono, H. (2019). Pemanfaatan Teknologi Informasi dalam Penilaian Hasil Belajar pada Generasi Milenial di Era Revolusi Industri 4.0. *Proceeding of Biology Education*, 5.
- Walters, A. (2022, Mei 2022). Retrieved from *testRigor*: <https://testrigor.com/blog/record-and-playback-with-selenium-ide-how-do-you-implement-it/>
- Wibowo, K. M., Kanedi, I., & Jumadi, J. (2015). SISTEM INFORMASI GEOGRAFIS (SIG) MENENTUKAN LOKASI PERTAMBANGAN BATU BARA DI PROVINSI BENGKULU BERBASIS WEBSITE. *Jurnal Media Infotama* , 5.
- Wijaya, A., Hendrastuty, N., Damayanti, & An'ars, M. G. (2022). RANCANG BANGUN SISTEM INFORMASI MANAJEMEN KEPEGAWAIAN

(SIMPEG) BERBASIS WEB (STUDI KASUS: PT SEMBILAN HAKIM NUSANTARA). *Jurnal Teknologi dan Sistem Informasi*, 2.