

**PENGEMBANGAN SISTEM *SELF-ORDER* BERBASIS
QR CODE DENGAN MANAJEMEN SESI MEJA
PADA USAHA KULINER**



Disusun Oleh:

N a m a : Adin Abimanyu
NIM : 22523060

**PROGRAM STUDI INFORMATIKA – PROGRAM SARJANA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA**

2026

HALAMAN PENGESAHAN DOSEN PEMBIMBING

**PENGEMBANGAN SISTEM *SELF-ORDER* BERBASIS
QR CODE DENGAN MANAJEMEN SESI MEJA
PADA USAHA KULINER**

TUGAS AKHIR JALUR MAGANG



Pembimbing,



(Prof., Dr. Sri Kusumadewi, S.Si., M.T.)

HALAMAN PENGESAHAN DOSEN PENGUJI

**PENGEMBANGAN SISTEM *SELF-ORDER* BERBASIS
QR CODE DENGAN MANAJEMEN SESI MEJA
PADA USAHA KULINER**

TUGAS AKHIR JALUR MAGANG

Telah dipertahankan di depan sidang penguji sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer dari Program Studi Informatika – Program Sarjana di Fakultas Teknologi Industri Universitas Islam Indonesia

Yogyakarta, 24 Juli 2026

Tim Penguji

Prof., Dr. Sri Kusumadewi, S.Si., M.T.

Anggota 1

Dr. Nur Wijayaning Rahayu, S.Kom., M.Cs.

Anggota 2

Fayruz Rahma, S.T., M.Eng.

Mengetahui,

Ketua Program Studi Informatika – Program Sarjana

Fakultas Teknologi Industri

Universitas Islam Indonesia

(Ir. Dhomas Hatta Fudholi, S.T., M.Eng., Ph.D. IPM., ASEAN Eng.)

HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan di bawah ini:

Nama : Adin Abimanyu
NIM : 22523060

Tugas akhir dengan judul:

PENGEMBANGAN SISTEM *SELF-ORDER* BERBASIS *QR CODE* DENGAN MANAJEMEN SESI MEJA PADA USAHA KULINER

Menyatakan bahwa seluruh komponen dan isi dalam tugas akhir ini adalah hasil karya saya sendiri. Apabila di kemudian hari terbukti ada beberapa bagian dari karya ini adalah bukan hasil karya sendiri, tugas akhir yang diajukan sebagai hasil karya sendiri ini siap ditarik kembali dan siap menanggung risiko dan konsekuensi apapun.

Demikian surat pernyataan ini dibuat, semoga dapat dipergunakan sebagaimana mestinya.

Yogyakarta, 28 Juni 2026



(Adin Abimanyu)

HALAMAN PERSEMBAHAN

Alhamdulillahirabbil'alamin, puji syukur penulis panjatkan kepada Allah Swt. atas segala rahmat, nikmat, dan kemudahan yang diberikan sehingga laporan Tugas Akhir Jalur Magang ini dapat diselesaikan dengan baik. Shalawat serta salam senantiasa tercurah kepada Nabi Muhammad saw.

Laporan ini penulis persembahkan kepada kedua orang tua dan keluarga yang selalu memberikan doa, dukungan, perhatian, serta kepercayaan kepada penulis selama menempuh pendidikan hingga penyusunan tugas akhir ini. Terima kasih atas segala pengorbanan, nasihat, dan dukungan yang diberikan selama ini.

Penulis juga mempersembahkan laporan ini kepada Ibu Prof., Dr. Sri Kusumadewi, S.Si., M.T. selaku dosen pembimbing, serta supervisor, mentor, dan rekan kerja di PT Vastu Cipta Persada yang telah memberikan bimbingan, arahan, dukungan, serta banyak pengalaman dan pembelajaran yang berharga selama pelaksanaan magang hingga penyusunan tugas akhir ini.

Terakhir, penulis mempersembahkan laporan ini kepada teman-teman yang telah memberikan bantuan, dukungan, dan kebersamaan selama masa perkuliahan hingga penyusunan tugas akhir ini. Semoga segala kebaikan yang telah diberikan mendapatkan balasan yang terbaik dari Allah Swt.

HALAMAN MOTO

“Aku membahayakan nyawa ibu untuk lahir ke dunia, jadi tidak mungkin aku tidak ada artinya. Aku membuat ayah bekerja setiap hari hingga lelah, jadi aku pastikan lelahnya tidak sia-sia.”

KATA PENGANTAR

Assalamu'alaikum Warahmatullahi Wabarakatuh.

Alhamdulillahirabbil'amin, segala puji dan syukur penulis panjatkan ke hadirat Allah Swt. atas segala rahmat, karunia, dan hidayah-Nya sehingga penulis dapat menyelesaikan Laporan Tugas Akhir Jalur Magang yang berjudul "Pengembangan Sistem *Self-Order* Berbasis *QR Code* dengan Manajemen Sesi Meja pada Usaha Kuliner" dengan baik.

Laporan ini disusun sebagai salah satu syarat untuk menyelesaikan Program Studi Informatika – Program Sarjana, Fakultas Teknologi Industri, Universitas Islam Indonesia melalui jalur magang. Laporan ini juga disusun sebagai bentuk dokumentasi dan pertanggungjawaban atas kegiatan magang yang telah dilaksanakan di PT Vastu Cipta Persada, khususnya pada tim pengembangan sistem Vastech.

Selama pelaksanaan magang dan penyusunan laporan ini, penulis memperoleh banyak pengalaman, pengetahuan, serta bantuan dari berbagai pihak. Oleh karena itu, pada kesempatan ini penulis menyampaikan ucapan terima kasih kepada:

1. Allah Swt. yang senantiasa memberikan kesehatan, kemudahan, dan kelancaran sehingga penulis dapat menyelesaikan kegiatan magang dan penyusunan laporan tugas akhir ini.
2. Kedua orang tua dan keluarga penulis yang selalu memberikan doa, dukungan, motivasi, serta semangat selama masa perkuliahan hingga penyusunan laporan tugas akhir.
3. Ibu Prof., Dr. Sri Kusumadewi, S.Si., M.T. selaku dosen pembimbing yang telah memberikan arahan, masukan, dan bimbingan selama proses penyusunan laporan tugas akhir.
4. Bapak dan Ibu dosen Program Studi Informatika – Program Sarjana Universitas Islam Indonesia yang telah memberikan ilmu dan pengalaman selama masa perkuliahan.
5. Direktur PT Vastu Cipta Persada beserta jajarannya yang telah memberikan kesempatan kepada penulis untuk melaksanakan kegiatan magang.
6. Supervisor, mentor, dan seluruh rekan kerja di PT Vastu Cipta Persada yang telah memberikan bimbingan, pengalaman kerja, serta kesempatan untuk terlibat dalam proses pengembangan sistem.
7. Diosi Siska Uli Putri Ananda Lubis, yang selalu memberikan dukungan, perhatian, serta semangat kepada penulis selama proses penyusunan laporan tugas akhir ini.

Terima kasih karena selalu meluangkan waktu untuk menemani penulis selama proses pengerjaan, serta memberikan semangat ketika penulis merasa lelah. Dukungan yang diberikan menjadi salah satu penyemangat bagi penulis untuk menyelesaikan tugas akhir ini dengan baik.

8. Teman-teman dan seluruh pihak yang telah memberikan bantuan, dukungan, dan motivasi selama pelaksanaan magang maupun penyusunan laporan tugas akhir.

Penulis menyadari bahwa laporan ini masih belum sempurna. Oleh karena itu, berbagai masukan dan saran sangat penulis harapkan untuk penyempurnaan laporan ini. Semoga laporan ini dapat memberikan manfaat bagi penulis dan pembaca lainnya.

Yogyakarta, 28 Juni 2026

A handwritten signature in black ink, consisting of a large, stylized 'A' followed by several loops and a final vertical stroke.

(Adin Abimanyu)

SARI

Penggunaan teknologi digital pada usaha kuliner terus berkembang untuk meningkatkan kualitas pelayanan dan efisiensi operasional. Salah satu bentuk penerapannya adalah sistem *self-order* yang memungkinkan pelanggan melakukan pemesanan secara mandiri melalui perangkat *smartphone*. Namun, penerapan layanan tersebut perlu didukung dengan pengelolaan penggunaan meja yang baik agar proses pemesanan dapat berlangsung sesuai dengan kondisi operasional restoran.

Selama melaksanakan kegiatan magang di PT Vastu Cipta Persada, penulis bergabung dengan tim Vastech yang bertanggung jawab dalam pengembangan sistem. Pada kegiatan magang ini, penulis terlibat dalam pengembangan sistem *Point of Sale* (POS) bernama CAVAA, khususnya pada modul *customer*. Pengembangan yang dilakukan berfokus pada fitur *self-order* berbasis *QR Code* dan manajemen sesi meja untuk mendukung proses pemesanan makanan dan minuman secara mandiri pada usaha kuliner.

Pengembangan sistem dilakukan menggunakan metode *Agile* dengan kerangka kerja *Scrum*. Teknologi yang digunakan meliputi PHP, Laravel, MySQL, JavaScript, Bootstrap, dan Git. Fitur yang dikembangkan mencakup autentikasi *customer*, implementasi *QR Code*, manajemen sesi meja, halaman menu, keranjang pesanan, *checkout*, detail pesanan, serta riwayat pesanan. Sistem juga terintegrasi dengan layanan pembayaran digital melalui *payment gateway*.

Hasil pengembangan menunjukkan bahwa sistem berhasil mendukung proses pemesanan mandiri pada usaha kuliner. Pelanggan dapat melakukan pemesanan melalui pemindaian *QR Code* pada meja yang tersedia, sementara manajemen sesi meja membantu mengatur kondisi penggunaan meja sehingga proses pemesanan dapat berjalan sesuai ketentuan yang berlaku. Sistem yang dikembangkan diharapkan dapat membantu meningkatkan efektivitas pelayanan, mengurangi kesalahan pencatatan pesanan, serta mendukung operasional usaha kuliner secara lebih terstruktur.

Kata kunci: *Agile*, Laravel, manajemen sesi meja, *Point of Sale*, *QR Code*, *Scrum*, *self-order*.

GLOSARIUM

<i>Agile</i>	Pendekatan pengembangan perangkat lunak yang dilakukan secara bertahap dan fleksibel sehingga perubahan kebutuhan dapat disesuaikan selama proses pengembangan berlangsung.
Bootstrap	<i>Framework</i> CSS yang digunakan untuk membantu pembuatan tampilan web agar lebih responsif dan dapat menyesuaikan berbagai ukuran layar perangkat.
Git	Alat yang digunakan untuk mengelola perubahan kode program selama proses pengembangan.
Laravel	<i>Framework</i> PHP yang digunakan dalam pengembangan aplikasi web untuk membantu pengelolaan logika sistem, <i>routing</i> , dan akses data.
MySQL	Sistem manajemen basis data yang digunakan untuk menyimpan dan mengelola data yang diperlukan oleh sistem.
<i>Point of Sale</i>	Sistem yang digunakan untuk membantu proses transaksi dan pengelolaan operasional pada usaha kuliner.
<i>QR Code</i>	Kode dua dimensi yang dapat dipindai untuk mengakses halaman atau informasi tertentu pada sistem.
<i>Scrum</i>	Kerangka kerja dalam metode <i>Agile</i> yang membagi proses pengembangan ke dalam beberapa <i>sprint</i> agar pekerjaan lebih terstruktur.
<i>Self-Order</i>	Layanan yang memungkinkan pelanggan melakukan pemesanan makanan dan minuman secara mandiri.
<i>Sprint</i>	Tahapan kerja dalam <i>Scrum</i> yang digunakan untuk menyelesaikan sejumlah pekerjaan dalam waktu yang telah ditentukan.

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PENGESAHAN DOSEN PEMBIMBING.....	ii
HALAMAN PENGESAHAN DOSEN PENGUJI.....	iii
HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR	iv
HALAMAN PERSEMBAHAN	v
HALAMAN MOTO	vi
KATA PENGANTAR	vii
SARI	ix
GLOSARIUM.....	x
DAFTAR ISI.....	xi
DAFTAR TABEL.....	xiii
DAFTAR GAMBAR.....	xiv
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Ruang Lingkup	2
1.3 Tujuan.....	2
1.4 Manfaat.....	3
1.5 Sistematika Penulisan.....	3
BAB II LANDASAN TEORI DAN TINJAUAN PUSTAKA	5
2.1 Sistem Informasi.....	5
2.2 <i>Point of Sale (POS)</i>	6
2.3 <i>Self-Order</i>	6
2.4 <i>Quick Response Code (QR Code)</i>	7
2.5 Manajemen Sesi Meja	7
2.6 <i>Agile Software Development</i>	8
2.7 <i>Scrum</i>	8
2.8 <i>System Usability Scale (SUS)</i>	9
2.9 Tinjauan Pustaka	10
BAB III PELAKSANAAN MAGANG.....	12
3.1 Manajemen Proyek.....	12
3.1.1 Gambaran Proyek	12
3.1.2 Alur Pelaksanaan Proyek.....	14
3.1.3 Pelaksanaan Proyek	15

3.2	Analisis dan Perancangan Sistem	17
3.2.1	Analisis Kebutuhan Sistem	18
3.2.2	<i>Use Case Diagram</i>	19
3.2.3	<i>Activity Diagram</i>	20
3.2.4	<i>Entity Relationship Diagram (ERD)</i>	21
3.2.5	Arsitektur Sistem	22
3.3	Implementasi Fitur	23
3.3.1	Teknologi yang Digunakan	24
3.3.2	Persiapan Lingkungan Pengembangan	24
3.3.3	Autentikasi <i>Customer</i>	28
3.3.4	Implementasi <i>QR Code</i> dan Manajemen Sesi Meja	32
3.3.5	Halaman <i>Customer</i>	34
3.3.6	Keranjang Pesanan dan <i>Checkout</i>	39
3.3.7	Halaman Detail Pesanan dan Riwayat Pesanan	43
3.4	Pengujian Sistem	46
3.4.1	Pengujian Fungsional Sistem	46
3.4.2	Pengujian <i>Usability</i> Menggunakan <i>System Usability Scale (SUS)</i>	47
BAB IV REFLEKSI PELAKSANAAN MAGANG		50
4.1	Relevansi Akademik	50
4.2	Pembelajaran Magang	51
BAB V PENUTUP		54
5.1	Kesimpulan	54
5.2	Saran	54
DAFTAR PUSTAKA		56
LAMPIRAN		58

DAFTAR TABEL

Tabel 2.1 Perbandingan Fitur Sistem.....	11
Tabel 3.1 Pembagian Pekerjaan pada Setiap <i>Sprint</i>	16
Tabel 3.2 Hasil <i>Scrum Meeting</i> pada Setiap <i>Sprint</i>	16
Tabel 3.3 Kebutuhan Fungsional Sistem	18
Tabel 3.4 Kebutuhan Nonfungsional Sistem	19
Tabel 3.5 Teknologi yang Digunakan dalam Pengembangan Sistem.....	24
Tabel 3.6 Hasil Pengujian Fungsional Sistem	47
Tabel 3.7 Hasil Pengujian SUS.....	48

DAFTAR GAMBAR

Gambar 3.1 Alur Pelaksanaan Proyek Pengembangan Sistem <i>Self-Order</i>	15
Gambar 3.2 <i>Use Case Diagram</i> Sistem <i>Self-Order</i>	20
Gambar 3.3 <i>Activity Diagram</i> proses pemesanan	21
Gambar 3.4 <i>Entity Relationship Diagram</i> Sistem <i>Self-Order</i>	22
Gambar 3.5 Arsitektur sistem <i>self-order</i>	23
Gambar 3.6 <i>Branch</i> Pengembangan Proyek CAVAA	25
Gambar 3.7 Struktur Proyek CAVAA	26
Gambar 3.8 Konfigurasi Sistem pada File <i>.env</i>	27
Gambar 3.9 Sistem CAVAA Berhasil Dijalankan	27
Gambar 3.10 Halaman Pilihan <i>Login Customer</i>	28
Gambar 3.11 Halaman <i>Login Customer</i>	29
Gambar 3.12 Halaman Registrasi <i>Customer</i>	30
Gambar 3.13 Halaman Lupa <i>Password</i>	31
Gambar 3.14 Penyimpanan Informasi <i>Outlet</i> dan Meja pada <i>Session</i>	31
Gambar 3.15 <i>Login</i> Otomatis Setelah Reset <i>Password</i>	32
Gambar 3.16 Halaman Pengelolaan <i>QR Code</i> Meja	33
Gambar 3.17 Validasi Status Meja	33
Gambar 3.18 Halaman Status Meja Tidak Tersedia	34
Gambar 3.19 Halaman Menu <i>Customer</i>	35
Gambar 3.20 Kategori Produk	36
Gambar 3.21 Halaman Detail Produk	37
Gambar 3.22 Pengambilan Data Produk	37
Gambar 3.23 Penyimpanan Data Keranjang pada <i>Browser</i>	39
Gambar 3.24 Halaman Keranjang Pesanan	40
Gambar 3.25 Halaman <i>Checkout</i>	41
Gambar 3.26 Validasi Ketersediaan Stok Saat <i>Checkout</i>	42
Gambar 3.27 Pembuatan Data Transaksi Pesanan	42
Gambar 3.28 Informasi Transaksi dan <i>QR Code</i> Pembayaran	43
Gambar 3.29 <i>QR Code</i> Transaksi	44
Gambar 3.30 Detail Produk Pesanan	44
Gambar 3.31 Halaman Riwayat Pesanan	45
Gambar 3.32 Pengambilan Data Riwayat Pesanan	45

Gambar 3.33 Navigasi ke Halaman Detail Pesanan	46
--	----

BAB I

PENDAHULUAN

1.1 Latar Belakang

PT Vastu Cipta Persada merupakan perusahaan yang bergerak di bidang perencanaan dan perancangan konstruksi bangunan. Selain menjalankan kegiatan pada sektor konstruksi, perusahaan juga memiliki divisi teknologi informasi bernama Vastech yang bertugas mengembangkan serta mengelola berbagai sistem yang digunakan perusahaan maupun klien. Salah satu produk yang dikembangkan oleh tim tersebut adalah sistem *Point of Sale* (POS) CAVAA yang dibuat untuk mendukung kebutuhan operasional usaha kuliner. Selama kegiatan magang berlangsung, pengembangan sistem CAVAA, khususnya pada fitur yang berhubungan dengan layanan pemesanan dan aktivitas pelanggan di restoran, menjadi bagian dari kegiatan yang dilaksanakan.

Meskipun berbagai teknologi telah diterapkan pada usaha kuliner, proses pemesanan pada beberapa restoran masih bergantung pada bantuan pelayan. Pelanggan harus menunggu pelayan untuk memperoleh informasi menu maupun menyampaikan pesanan yang ingin dilakukan. Kondisi tersebut dapat menyebabkan antrian pelanggan, keterlambatan penyampaian pesanan ke dapur, serta kesalahan dalam pencatatan pesanan. Dampak dari permasalahan tersebut dapat memengaruhi kualitas pelayanan dan kenyamanan pelanggan selama berada di restoran (Pudyawardana, 2023).

Penggunaan *QR Code* mulai diterapkan sebagai media akses menuju menu digital. Melalui proses pemindaian menggunakan *smartphone*, pelanggan dapat langsung melihat daftar menu tanpa harus menggunakan buku menu fisik (Rusito & Diyah Intan Mawarni Putri, 2024). *QR Code* dapat mendukung penerapan layanan *self-order* yang memungkinkan pelanggan melakukan pemesanan secara mandiri. Pelanggan dapat melihat menu, memilih produk, menentukan jumlah pesanan, memasukkan produk ke keranjang, hingga menyelesaikan proses *checkout* melalui sistem (Dzikria & Rizal, 2023). Meskipun demikian, penerapan layanan *self-order* tetap memerlukan pengelolaan penggunaan meja yang baik. Setiap meja perlu memiliki informasi status yang jelas agar hanya meja yang memenuhi ketentuan penggunaan yang dapat digunakan untuk melakukan transaksi. Tanpa adanya mekanisme tersebut, pelanggan berpotensi mengakses halaman pemesanan pada meja yang sebenarnya sedang tidak tersedia atau tidak dapat digunakan. Kondisi tersebut dapat menyebabkan ketidaksesuaian informasi yang berdampak pada proses operasional restoran.

Oleh sebab itu, diperlukan sebuah sistem yang mampu mengintegrasikan layanan *self-order* berbasis *QR Code* dengan manajemen sesi meja sehingga proses pemesanan dapat berlangsung sesuai kondisi penggunaan meja yang berlaku (Rezeki & Alda, 2024).

Berdasarkan permasalahan tersebut, tugas akhir ini membahas pengembangan sistem *self-order* berbasis *QR Code* yang dilengkapi fitur manajemen sesi meja pada usaha kuliner. Sistem yang dikembangkan memungkinkan pelanggan melakukan pemesanan makanan dan minuman secara mandiri melalui *smartphone* setelah memindai *QR Code* yang tersedia pada meja. Sistem ini dilengkapi mekanisme pengelolaan sesi meja untuk mengatur penggunaan meja sesuai kondisi operasional yang berlaku. Sistem mendukung beberapa metode pembayaran, termasuk pembayaran digital yang terhubung dengan layanan *payment gateway*. Melalui penerapan sistem tersebut, proses pemesanan diharapkan menjadi lebih efektif, meminimalkan kesalahan pencatatan pesanan, serta membantu meningkatkan kualitas pelayanan pada usaha kuliner.

1.2 Ruang Lingkup

Kegiatan magang dilaksanakan selama enam bulan, yaitu mulai 15 September 2025 sampai 13 Maret 2026 di PT Vastu Cipta Persada. Selama kegiatan tersebut, pengembangan sistem *Point of Sale* (POS) CAVAA menjadi bagian dari kegiatan magang untuk mendukung operasional usaha kuliner. Fokus pekerjaan selama kegiatan magang adalah pengembangan fitur *self-order* berbasis *QR Code* yang terintegrasi dengan manajemen sesi meja pada modul *customer*. Fitur tersebut dirancang agar pelanggan dapat melakukan pemesanan makanan dan minuman secara mandiri menggunakan *smartphone*.

Selain pengembangan fitur, kegiatan magang juga mencakup proses implementasi sistem, perbaikan fungsi, serta pengujian terhadap fitur yang telah dikembangkan. Karena sistem CAVAA memiliki cakupan yang cukup luas dan terdiri dari berbagai modul yang saling terintegrasi, laporan tugas akhir ini difokuskan pada pengembangan fitur *self-order* berbasis *QR Code* dan manajemen sesi meja pada modul *customer*.

1.3 Tujuan

Pengembangan sistem *self-order* berbasis *QR Code* dengan manajemen sesi meja pada usaha kuliner memiliki beberapa tujuan sebagai berikut:

- a. Mengembangkan layanan *self-order* berbasis *QR Code* pada sistem CAVAA agar pelanggan dapat mengakses menu dan melakukan pemesanan secara mandiri melalui *smartphone*.
- b. Mengembangkan fitur manajemen sesi meja yang digunakan untuk mengatur kondisi penggunaan meja sesuai status yang ditetapkan pada sistem.
- c. Mengembangkan fitur pemesanan pada modul *customer* yang mencakup proses pemilihan menu, pengelolaan keranjang pesanan, dan *checkout*.
- d. Mengembangkan fitur detail pesanan dan riwayat pesanan sebagai sarana bagi pelanggan untuk memantau informasi transaksi yang dilakukan melalui sistem.

1.4 Manfaat

Pengembangan sistem *self-order* berbasis *QR Code* dengan manajemen sesi meja diharapkan dapat memberikan beberapa manfaat sebagai berikut:

- a. Membantu usaha kuliner mengelola proses pemesanan dan penggunaan meja secara lebih teratur.
- b. Memberikan kemudahan bagi pelanggan dalam melakukan pemesanan tanpa harus menunggu bantuan pelayan.
- c. Membantu penyampaian informasi pesanan menjadi lebih cepat.
- d. Mengurangi antrian panjang dan membuat pelanggan merasa nyaman selama proses pemesanan berlangsung.

1.5 Sistematika Penulisan

Sistematika penulisan laporan tugas akhir jalur magang ini disusun menjadi beberapa bab agar pembahasan dapat disampaikan secara terstruktur dan mudah dipahami. Adapun susunannya sebagai berikut:

- a. BAB I: Pendahuluan

Bab ini menjelaskan latar belakang, ruang lingkup, tujuan, manfaat, serta sistematika penulisan.

- b. BAB II: Landasan Teori dan Tinjauan Pustaka

Bab ini berupa pengertian dan definisi yang berkaitan dengan penyusunan laporan.

- c. BAB III: Pelaksanaan Magang

Bab ini menjelaskan mengenai proses pelaksanaan magang dalam pengembangan sistem.

d. BAB IV: Refleksi Pelaksanaan Magang

Bab ini berisi pembelajaran yang diperoleh selama melaksanakan magang, relevansi kegiatan magang terhadap bidang akademik, serta pengalaman yang didapatkan selama bekerja di lingkungan profesional.

e. BAB V: Penutup

Bab ini berisi kesimpulan yang didapatkan dari pembahasan yang telah dilakukan pada bab-bab sebelumnya serta saran yang dapat digunakan untuk pengembangan sistem pada masa mendatang.

BAB II

LANDASAN TEORI DAN TINJAUAN PUSTAKA

Landasan teori dan tinjauan pustaka menjadi dasar dalam pengembangan sistem. Beberapa konsep yang berkaitan dengan sistem yang dikembangkan dijelaskan pada bagian ini, antara lain sistem informasi, *Point of Sale*, *self-order*, *QR Code*, manajemen sesi meja, *Agile Software Development*, *Scrum*, dan *System Usability Scale*. Beberapa penelitian terdahulu yang relevan juga disajikan sebagai bahan perbandingan untuk menunjukkan perbedaan serta kontribusi sistem yang dikembangkan.

2.1 Sistem Informasi

Setiap kegiatan yang dilakukan dalam suatu organisasi menghasilkan berbagai data yang perlu dikelola dengan baik agar dapat dimanfaatkan sebagai informasi. Informasi tersebut memiliki peran penting dalam mendukung pelaksanaan kegiatan operasional maupun membantu proses pengambilan keputusan. Untuk mengelola data tersebut, diperlukan sistem informasi yang mampu menghubungkan beberapa komponen, seperti perangkat keras, perangkat lunak, data, prosedur kerja, dan pengguna dalam satu proses yang saling berkaitan (Sandfreni et al., 2021).

Masing-masing komponen dalam sistem informasi memiliki fungsi yang berbeda, tetapi saling mendukung satu sama lain. Perangkat keras digunakan sebagai sarana untuk menjalankan sistem, sedangkan perangkat lunak bertugas mengolah data sesuai kebutuhan. Data yang tersimpan menjadi sumber utama informasi, sementara prosedur kerja dan pengguna berperan dalam memastikan seluruh proses dapat berjalan sesuai tujuan organisasi (Asher & Hidayat, 2024).

Keberadaan sistem informasi membantu organisasi mengelola data secara lebih teratur dibandingkan proses yang dilakukan secara manual. Berbagai aktivitas pengolahan data dapat dilakukan dengan lebih cepat sehingga pekerjaan menjadi lebih efisien dan kemungkinan terjadinya kesalahan dapat dikurangi. Informasi yang telah tersimpan dalam sistem juga lebih mudah diakses ketika diperlukan karena seluruh data tersusun secara terorganisasi dalam satu tempat (Sandfreni et al., 2021).

2.2 *Point of Sale (POS)*

Dalam kegiatan bisnis, proses transaksi menjadi salah satu aktivitas utama yang perlu dikelola dengan baik. Untuk mendukung kegiatan tersebut, banyak usaha memanfaatkan sistem *Point of Sale (POS)* sebagai sarana pencatatan dan pengelolaan transaksi penjualan. Melalui sistem ini, data transaksi dapat direkam secara otomatis, termasuk pengelolaan data produk, perhitungan pembayaran, serta penyimpanan riwayat transaksi yang telah dilakukan. Seiring perkembangan teknologi, fungsi POS tidak hanya terbatas pada pencatatan transaksi, tetapi juga mendukung berbagai kebutuhan operasional lainnya (Juventauricula et al., 2024).

Pada usaha kuliner, sistem POS membantu proses pelayanan menjadi lebih teratur karena setiap pesanan yang diterima dapat langsung tercatat dalam sistem. Kondisi tersebut memudahkan pengelolaan informasi pesanan sekaligus membantu memantau aktivitas penjualan yang berlangsung. Sistem POS juga dapat digunakan untuk mengelola menu serta menyediakan berbagai informasi yang diperlukan dalam kegiatan operasional sehari-hari (Pudyawardana, 2023).

Penggunaan sistem POS memberikan sejumlah keuntungan bagi pelaku usaha. Proses transaksi dapat dilakukan dengan lebih cepat, pencatatan data menjadi lebih rapi, dan informasi penjualan lebih mudah dipantau. Dengan adanya sistem tersebut, data yang berkaitan dengan operasional usaha dapat tersimpan secara lebih terstruktur sehingga dapat diakses kembali ketika dibutuhkan (Asra et al., 2023).

2.3 *Self-Order*

Self-order merupakan layanan yang memungkinkan pelanggan melakukan pemesanan secara mandiri tanpa harus bergantung pada bantuan pelayan (Dzikria & Rizal, 2023). Melalui layanan ini, pelanggan dapat melihat menu yang tersedia, memilih produk yang diinginkan, menentukan jumlah pesanan, hingga melakukan konfirmasi pesanan melalui perangkat yang digunakan.

Penggunaan layanan *self-order* semakin banyak diterapkan pada usaha kuliner karena dapat membantu mempercepat proses pelayanan. Pelanggan tidak perlu menunggu pelayan datang ke meja untuk melakukan pemesanan sehingga proses pemilihan menu dan pemesanan dapat dilakukan secara langsung sesuai kebutuhan. Data pesanan yang masuk akan tersimpan secara otomatis ke dalam sistem sehingga kemungkinan terjadinya kesalahan akibat pencatatan manual dapat dikurangi (Susila & Sri Arsa, 2023).

Kemudahan penggunaan juga menjadi salah satu alasan layanan *self-order* banyak digunakan. Dengan memanfaatkan *smartphone*, pelanggan dapat melakukan pemesanan kapan saja selama layanan tersedia. Proses pemesanan yang dilakukan secara mandiri membantu menciptakan alur pelayanan yang lebih cepat dan lebih teratur dibandingkan metode pemesanan secara manual (Yoseph Halim et al., 2021).

2.4 Quick Response Code (QR Code)

Quick Response Code atau *QR Code* merupakan bentuk kode dua dimensi yang digunakan untuk menyimpan berbagai informasi dalam bentuk pola tertentu yang dapat dibaca menggunakan perangkat digital (Rusito & Diah Intan Mawarni Putri, 2024). Teknologi ini dikembangkan untuk memudahkan proses penyimpanan dan penyampaian informasi secara cepat melalui proses pemindaian.

Saat ini, penggunaan *QR Code* telah diterapkan pada berbagai bidang, termasuk sektor kuliner. Pengguna hanya perlu memindai kode menggunakan kamera *smartphone* untuk mengakses informasi yang tersimpan di dalamnya. Informasi tersebut dapat berupa alamat web, informasi produk, data transaksi, maupun informasi lainnya sesuai kebutuhan sistem (Anggoro et al., 2021).

Pada usaha kuliner, *QR Code* sering dimanfaatkan sebagai media akses menuju menu digital dan layanan pemesanan. Penggunaan *QR Code* membantu pelanggan memperoleh informasi dengan lebih mudah tanpa perlu mengetik alamat web secara manual. Penggunaan menu digital juga dapat mengurangi ketergantungan terhadap media cetak sehingga informasi menu lebih mudah diperbarui apabila terjadi perubahan (Rusito & Diah Intan Mawarni Putri, 2024).

2.5 Manajemen Sesi Meja

Manajemen sesi meja merupakan mekanisme yang digunakan untuk mengatur kondisi penggunaan meja dalam suatu sistem pemesanan. Melalui mekanisme tersebut, setiap meja dapat memiliki status tertentu yang menunjukkan kondisi penggunaannya pada waktu tertentu (Kudadiri, 2025).

Pada operasional restoran, pengelolaan meja menjadi bagian yang perlu diperhatikan karena berkaitan langsung dengan proses pelayanan kepada pelanggan. Informasi mengenai kondisi meja harus tersedia secara jelas agar proses pemesanan dapat dilakukan sesuai aturan yang berlaku. Pengelolaan yang baik juga membantu mengurangi kemungkinan terjadinya

penggunaan meja yang tidak sesuai dengan kondisi operasional restoran (Wijaya & Wahyuningtyas, 2024).

Penerapan manajemen sesi meja memungkinkan sistem membedakan meja yang sedang digunakan, tersedia, maupun berada pada kondisi tertentu sesuai kebutuhan restoran. Dengan adanya pengaturan tersebut, proses pemesanan dapat berlangsung dengan lebih terkontrol. Selain membantu pengelolaan meja, mekanisme ini juga mendukung kelancaran pelayanan dan membantu operasional restoran berjalan lebih teratur (Rezeki & Alda, 2024).

2.6 Agile Software Development

Dalam proses pengembangan perangkat lunak, perubahan kebutuhan sering terjadi selama pengerjaan sistem berlangsung. Kondisi tersebut membuat tim pengembang perlu memiliki kemampuan untuk menyesuaikan proses kerja agar sistem yang dibangun tetap sesuai dengan kebutuhan pengguna. Salah satu pendekatan yang banyak digunakan untuk menghadapi kondisi tersebut adalah *Agile Software Development* (Nova et al., 2022).

Agile menerapkan proses pengembangan yang dilakukan secara bertahap dan berulang. Setiap tahap menghasilkan bagian sistem yang dapat diuji dan dievaluasi sebelum dilanjutkan ke tahap berikutnya. Melalui pendekatan tersebut, perubahan kebutuhan dapat ditangani dengan lebih mudah tanpa harus mengulang seluruh proses pengembangan dari awal (Saleh, 2023).

Agile juga mendorong komunikasi yang aktif antara anggota tim dan pihak yang terlibat dalam proyek. Komunikasi yang dilakukan secara rutin membantu tim memahami kebutuhan pengguna dengan lebih baik sehingga hasil pengembangan dapat lebih sesuai dengan tujuan yang ingin dicapai (Nova et al., 2022).

2.7 Scrum

Scrum merupakan salah satu kerangka kerja yang banyak digunakan dalam metode *Agile* untuk membantu tim mengelola proses pengembangan perangkat lunak. Melalui *Scrum*, pekerjaan dibagi menjadi beberapa periode pengerjaan yang disebut *sprint* sehingga proses pengembangan dapat dilakukan secara bertahap dan lebih mudah dipantau (Nilawati & Widya, 2023).

Setiap *sprint* memiliki target yang telah ditentukan sesuai kebutuhan sistem yang sedang dikembangkan. Pada akhir *sprint*, hasil pekerjaan akan ditinjau dan dievaluasi sebelum tim melanjutkan pengerjaan pada *sprint* berikutnya. Cara kerja tersebut membantu tim

menyesuaikan kebutuhan pengembangan tanpa harus mengulang pekerjaan yang telah diselesaikan sebelumnya (Walim et al., 2023).

Dalam pelaksanaannya, *Scrum* melibatkan beberapa peran utama, yaitu *Product Owner*, *Scrum Master*, dan *Development Team*. *Product Owner* bertanggung jawab menentukan kebutuhan yang akan dikembangkan dalam sistem. *Scrum Master* bertugas memastikan proses *Scrum* berjalan sesuai ketentuan yang telah ditetapkan. Sementara itu, *Development Team* bertanggung jawab menyelesaikan pekerjaan pengembangan sesuai tugas yang diberikan (Nilawati & Widya, 2023).

Scrum juga terdiri dari beberapa kegiatan utama, seperti *sprint planning*, *daily scrum*, *sprint review*, dan *sprint retrospective*. Kegiatan tersebut membantu tim menyusun rencana kerja, memantau perkembangan pengerjaan, melakukan evaluasi hasil *sprint*, serta memperbaiki proses kerja pada *sprint* berikutnya (Walim et al., 2023).

2.8 System Usability Scale (SUS)

System Usability Scale (SUS) merupakan metode yang digunakan untuk menilai tingkat kemudahan penggunaan suatu sistem berdasarkan pengalaman pengguna saat berinteraksi dengan sistem tersebut. Penilaian dilakukan melalui kuesioner yang berisi sepuluh pernyataan dengan rentang nilai 1 sampai 5. Nilai yang diberikan responden digunakan untuk menilai kemudahan penggunaan sistem setelah pengguna mencoba fitur-fitur yang tersedia (Susila & Sri Arsa, 2023).

Pada metode SUS, setiap jawaban responden terlebih dahulu diolah sesuai dengan ketentuan perhitungan yang telah ditetapkan. Pernyataan bernomor ganjil merupakan pernyataan positif sehingga skor dihitung dengan mengurangi nilai jawaban responden sebesar 1. Sebaliknya, pernyataan bernomor genap merupakan pernyataan negatif sehingga skor dihitung dari hasil pengurangan angka 5 dengan nilai jawaban responden. Seluruh skor yang diperoleh kemudian dijumlahkan. Karena kuesioner SUS terdiri atas sepuluh pernyataan, jumlah skor tersebut dikalikan dengan 2,5 sehingga menghasilkan nilai SUS pada rentang 0 hingga 100. Semakin tinggi nilai yang diperoleh, semakin baik tingkat kemudahan penggunaan sistem berdasarkan penilaian pengguna (Herawati & Azahra, 2024). Rumus yang digunakan untuk menghitung nilai SUS ditunjukkan pada Persamaan (2.1).

$$SUS = (\sum \text{skor}) \times 2,5 \quad (2.1)$$

Keterangan:

SUS = nilai akhir SUS

$\sum$ skor = total skor dari seluruh pernyataan kuesioner

2.9 Tinjauan Pustaka

Beberapa penelitian sebelumnya telah membahas penerapan sistem pemesanan pada restoran maupun usaha kuliner berbasis web. Penelitian yang dilakukan oleh (Pudyawardana, 2023) mengembangkan sistem pemesanan makanan berbasis web yang bertujuan membantu meningkatkan kualitas pelayanan restoran. Meskipun mampu mendukung proses pemesanan secara digital, sistem tersebut belum menyediakan layanan *self-order* maupun pengelolaan sesi meja.

Penelitian lain yang dilakukan oleh (Dzikria & Rizal, 2023) membahas pengembangan layanan pemesanan mandiri berbasis web. Sistem yang dikembangkan mampu membantu mempercepat proses pelayanan karena pelanggan dapat melakukan pemesanan secara langsung melalui sistem. Namun, penelitian tersebut belum mengintegrasikan pengelolaan kondisi meja dengan proses pemesanan yang dilakukan pelanggan. Selanjutnya, penelitian yang dilakukan oleh (Rusito & Diah Intan Mawarni Putri, 2024) membahas pemanfaatan *QR Code* pada sistem pemesanan restoran berbasis web. Hasil penelitian menunjukkan bahwa penggunaan *QR Code* memudahkan pelanggan dalam mengakses menu digital secara cepat. Akan tetapi, penelitian tersebut masih berfokus pada akses menu dan belum dilengkapi dengan fitur manajemen sesi meja. Di sisi lain, penelitian oleh (Rezeki & Alda, 2024) membahas sistem manajemen reservasi restoran yang mendukung pengelolaan meja dan status reservasi pelanggan. Meskipun telah menyediakan fitur yang berkaitan dengan pengelolaan meja, sistem tersebut lebih difokuskan pada proses reservasi dan belum mendukung layanan *self-order* secara menyeluruh. Perbandingan fitur yang terdapat pada penelitian terdahulu dengan sistem yang dikembangkan dapat dilihat pada Tabel 2.1.

Tabel 2.1 Perbandingan Fitur Sistem

No	Penelitian	Self Order	QR Code	Pemesanan Digital	Manajemen Sesi Meja
1	Pudyawardana (2023)	-	-	√	-
2	Dzikria & Rizal (2023)	√	-	√	-
3	Rusito & Putri (2024)	-	√	√	-
4	Rezeki & Alda (2024)	-	-	-	√
5	Sistem ini	√	√	√	√

Berdasarkan Tabel 2.1, dapat diketahui bahwa belum terdapat sistem yang menggabungkan fitur *self-order*, *QR Code*, pemesanan digital, dan manajemen sesi meja dalam satu layanan yang saling terhubung. Berdasarkan kondisi tersebut, dikembangkan sistem *self-order* berbasis *QR Code* dengan manajemen sesi meja untuk membantu meningkatkan efektivitas pelayanan dan pengelolaan operasional pada usaha kuliner.

BAB III

PELAKSANAAN MAGANG

Pelaksanaan magang dilakukan melalui beberapa tahapan yang saling berkaitan dalam pengembangan sistem. Pada bagian ini dijelaskan proses yang dilakukan selama kegiatan magang, meliputi manajemen proyek, analisis dan perancangan sistem, implementasi fitur, serta pengujian sistem sebagai tahap untuk memastikan fitur yang dikembangkan dapat berfungsi sesuai dengan kebutuhan.

3.1 Manajemen Proyek

Selama kegiatan magang, proses pengembangan sistem tidak hanya berfokus pada pembuatan fitur, tetapi juga membutuhkan pengelolaan pekerjaan yang terarah agar setiap target dapat dicapai sesuai rencana. Pengelolaan proyek membantu tim dalam mengatur urutan pekerjaan, menentukan prioritas pengembangan, membagi tugas, serta melakukan pemantauan terhadap progres yang sedang berjalan. Dengan adanya pengelolaan yang baik, setiap aktivitas pengembangan dapat dilaksanakan secara lebih terstruktur sehingga hambatan yang berpotensi mengganggu jalannya proyek dapat diminimalkan.

Pada pengembangan sistem CAVAA, setiap pekerjaan dilaksanakan melalui beberapa tahapan yang saling berkaitan. Proses dimulai dari memahami kebutuhan sistem, dilanjutkan dengan penyusunan rancangan, pengembangan fitur, pengujian, hingga penyempurnaan berdasarkan hasil evaluasi. Seluruh tahapan tersebut dilakukan melalui koordinasi bersama tim pengembang agar fitur yang dihasilkan dapat memenuhi kebutuhan pengguna dan mendukung tujuan pengembangan sistem. Oleh karena itu, manajemen proyek menjadi salah satu aspek yang berperan penting dalam menjaga kelancaran proses pengembangan selama pelaksanaan magang.

3.1.1 Gambaran Proyek

Selama menjalani program magang di PT Vastu Cipta Persada, penulis bergabung dengan tim pengembangan yang bertanggung jawab dalam pembuatan aplikasi berbasis web untuk kebutuhan perusahaan maupun klien. Salah satu proyek yang menjadi fokus selama periode magang adalah pengembangan sistem *Point of Sale* (POS) CAVAA yang digunakan untuk mendukung aktivitas operasional pada usaha kuliner.

CAVAA dikembangkan sebagai sistem yang menghubungkan berbagai kebutuhan operasional restoran dalam satu platform. Sistem ini terdiri atas beberapa modul, yaitu *admin*, *owner*, *partner (outlet)*, *kitchen*, *cashier*, dan *customer*. Masing-masing modul memiliki fungsi yang berbeda sesuai dengan peran penggunanya. Pada kegiatan magang ini, pengembangan difokuskan pada modul *customer* yang berhubungan langsung dengan aktivitas pelanggan saat melakukan pemesanan.

Pengembangan pada modul *customer* difokuskan pada penerapan layanan *self-order* berbasis *QR Code*. Melalui fitur tersebut, pelanggan dapat mengakses halaman pemesanan secara langsung menggunakan *smartphone* setelah melakukan pemindaian *QR Code* yang tersedia pada meja restoran. Dengan cara ini, pelanggan dapat melihat menu, memilih produk, serta melakukan pemesanan tanpa harus menunggu bantuan pelayan.

Sistem ini dilengkapi fitur manajemen sesi meja yang digunakan untuk mengontrol kondisi penggunaan meja. Fitur ini membantu sistem menentukan apakah suatu meja dapat digunakan untuk melakukan transaksi atau tidak berdasarkan status yang dimiliki. Pengaturan tersebut membantu menjaga kesesuaian informasi antara kondisi meja di lapangan dengan data yang tersimpan pada sistem. Adapun fitur yang dikembangkan selama pelaksanaan magang meliputi:

- a. Halaman Autentikasi yang digunakan pelanggan untuk melakukan proses *login* pada sistem.
- b. Halaman Menu yang menampilkan daftar makanan dan minuman yang tersedia untuk dipesan.
- c. Implementasi *QR Code* sebagai media akses menuju halaman pemesanan.
- d. Fitur Manajemen Sesi Meja yang digunakan untuk mengatur kondisi penggunaan meja berdasarkan status yang tersedia.
- e. Halaman Keranjang Pesanan yang menampilkan produk yang telah dipilih sebelum proses *checkout* dilakukan.
- f. Halaman *Checkout* yang digunakan untuk melakukan konfirmasi pesanan dan melanjutkan proses transaksi.
- g. Halaman Detail Pesanan yang menampilkan informasi dan perkembangan pesanan yang sedang diproses.
- h. Halaman Riwayat Pesanan yang digunakan untuk melihat kembali transaksi yang pernah dilakukan pelanggan.

CAVAA juga telah terintegrasi dengan beberapa metode pembayaran, termasuk pembayaran digital yang diproses melalui layanan *payment gateway*. Melalui pengembangan fitur-fitur tersebut, sistem diharapkan mampu membantu restoran dalam meningkatkan efisiensi pelayanan. Pelanggan dapat melakukan pemesanan dengan lebih praktis, sedangkan pihak restoran memperoleh kemudahan dalam mengelola pesanan dan memantau penggunaan meja melalui sistem yang terhubung secara terintegrasi.

3.1.2 Alur Pelaksanaan Proyek

Pengembangan sistem *self-order* berbasis *QR Code* dengan manajemen sesi meja dilakukan melalui beberapa tahapan yang saling berhubungan. Setiap tahapan memiliki peran dalam memastikan fitur yang dikembangkan dapat berjalan sesuai kebutuhan pengguna dan mendukung aktivitas operasional restoran. Selama kegiatan magang, penulis terlibat secara langsung mulai dari proses analisis kebutuhan hingga pengujian dan penyempurnaan sistem.

Tahap awal yang dilakukan adalah memahami kebutuhan sistem yang akan dikembangkan. Pada tahap ini, proses bisnis yang diterapkan pada sistem CAVAA dipelajari, struktur basis data yang digunakan dipahami, serta alur pemesanan yang sudah berjalan pada sistem ditinjau. Selain itu, dilakukan diskusi bersama tim pengembang untuk memperoleh gambaran mengenai fitur yang akan dibuat dan permasalahan yang ingin diselesaikan. Hasil dari kegiatan tersebut menjadi dasar dalam menentukan fungsi dan alur kerja sistem.

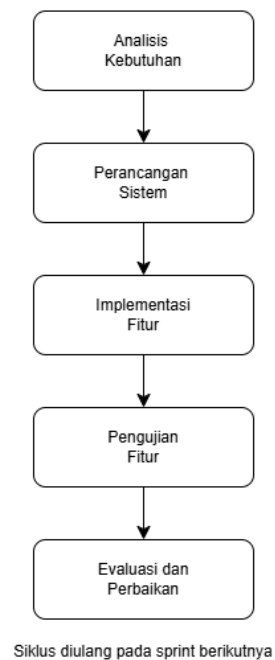
Setelah kebutuhan sistem dipahami, proses dilanjutkan dengan penyusunan rancangan sistem. Kegiatan yang dilakukan meliputi perancangan antarmuka pengguna, penyusunan alur penggunaan fitur, serta penyesuaian struktur data yang diperlukan. Rancangan tersebut digunakan sebagai panduan selama proses pengembangan sehingga implementasi yang dilakukan tetap sesuai dengan kebutuhan yang telah ditentukan sebelumnya.

Tahap selanjutnya adalah pengembangan fitur pada modul *customer*. Proses ini dilakukan secara bertahap sesuai prioritas pekerjaan yang telah ditetapkan. Pengembangan dimulai dari fitur autentikasi pengguna, implementasi *QR Code*, pengelolaan sesi meja, halaman menu, keranjang pesanan, *checkout*, detail pesanan, hingga riwayat transaksi pelanggan. Setiap fitur yang telah selesai dikembangkan terlebih dahulu diuji untuk memastikan fungsinya berjalan sesuai kebutuhan sebelum proses pengembangan dilanjutkan ke fitur berikutnya.

Setelah seluruh fitur utama selesai dikembangkan, dilakukan pengujian sistem secara menyeluruh untuk memastikan seluruh fitur dapat berjalan dengan baik dan saling terintegrasi.

Pengujian dilakukan menggunakan berbagai skenario yang menggambarkan kondisi penggunaan sistem secara nyata. Beberapa skenario yang diuji meliputi proses pemindaian *QR Code*, pengelolaan keranjang pesanan, pembuatan transaksi, serta proses pembayaran. Hasil pengujian digunakan sebagai bahan evaluasi untuk menemukan bagian sistem yang masih memerlukan perbaikan.

Tahap terakhir berupa evaluasi dan penyempurnaan sistem. Pada tahap ini, setiap kendala yang ditemukan selama pengujian diperbaiki agar sistem dapat berjalan dengan lebih stabil. Selain perbaikan *bug*, dilakukan juga beberapa penyesuaian berdasarkan masukan dari tim pengembang. Melalui proses tersebut, fitur yang dikembangkan dapat lebih sesuai dengan kebutuhan pengguna dan siap digunakan pada lingkungan operasional sistem. Alur pelaksanaan proyek dapat dilihat pada Gambar 3.1.



Gambar 3.1 Alur Pelaksanaan Proyek Pengembangan Sistem *Self-Order*

3.1.3 Pelaksanaan Proyek

Pengembangan sistem *self-order* berbasis *QR Code* dengan manajemen sesi meja dilakukan secara bertahap melalui beberapa *sprint*. Pembagian *sprint* digunakan untuk membantu tim mengatur pekerjaan agar proses pengembangan berjalan lebih terarah dan setiap fitur dapat dievaluasi sebelum memasuki pekerjaan berikutnya. Pembagian fitur yang dikerjakan selama proses pengembangan dapat dilihat pada Tabel 3.1.

Tabel 3.1 Pembagian Pekerjaan pada Setiap *Sprint*

Sprint	Fitur
Sprint 1	Perancangan basis data dan Pengembangan autentikasi <i>customer</i>
Sprint 2	Implementasi <i>QR Code</i> dan manajemen sesi meja
Sprint 3	Pengembangan halaman <i>customer</i>
Sprint 4	Pengembangan keranjang pesanan dan <i>checkout</i>
Sprint 5	Pengembangan halaman detail pesanan serta riwayat pesanan
Sprint 6	Pengujian sistem dan perbaikan <i>bug</i>

Berdasarkan pembagian *sprint* pada Tabel 3.1, proses pengembangan dilakukan secara bertahap sesuai target yang telah ditentukan bersama tim. Setiap *sprint* memiliki fokus pekerjaan yang berbeda sehingga progres pengembangan dapat dipantau dengan lebih mudah. Pembagian pekerjaan tersebut juga membantu tim dalam melakukan evaluasi pada setiap tahap pengerjaan sebelum melanjutkan ke *sprint* berikutnya.

Selama pelaksanaan proyek, *Scrum meeting* dilaksanakan secara berkala bersama supervisor dan anggota tim pada setiap *sprint*. Pertemuan tersebut digunakan untuk membahas perkembangan pekerjaan yang telah diselesaikan, kendala yang dihadapi selama proses pengembangan, serta menerima masukan dari supervisor maupun anggota tim terhadap fitur yang telah dikembangkan. Masukan yang diberikan pada setiap *Scrum meeting* menjadi bahan evaluasi untuk menyempurnakan fitur yang masih memerlukan penyesuaian sebelum melanjutkan pengembangan pada *sprint* berikutnya. Hasil *Scrum meeting* pada setiap *sprint* dapat dilihat pada Tabel 3.2.

Tabel 3.2 Hasil *Scrum Meeting* pada Setiap *Sprint*

Sprint	Hasil Scrum Meeting
Sprint 1	Hasil pengembangan autentikasi <i>customer</i> dibahas bersama supervisor dan anggota tim. Berdasarkan hasil diskusi, fitur autentikasi <i>customer</i> telah sesuai dengan kebutuhan sehingga pengembangan dapat dilanjutkan ke implementasi <i>QR Code</i> dan manajemen sesi meja pada <i>sprint</i> berikutnya.
Sprint 2	Hasil implementasi <i>QR Code</i> dan manajemen sesi meja dibahas bersama supervisor dan anggota tim. Berdasarkan hasil diskusi, dilakukan penyesuaian pada mekanisme manajemen sesi meja sebelum pengembangan dilanjutkan ke <i>sprint</i> berikutnya.
Sprint 3	Hasil pengembangan halaman <i>customer</i> , menu, kategori produk, dan detail produk dibahas bersama supervisor dan anggota tim. Berdasarkan hasil

	diskusi, pengembangan dapat dilanjutkan ke fitur keranjang pesanan dan <i>checkout</i> pada <i>sprint</i> berikutnya.
Sprint 4	Hasil pengembangan fitur keranjang pesanan dan <i>checkout</i> dibahas bersama supervisor dan anggota tim. Berdasarkan hasil diskusi, fitur keranjang pesanan dan <i>checkout</i> telah sesuai dengan kebutuhan sehingga pengembangan dapat dilanjutkan ke halaman detail pesanan dan riwayat pesanan.
Sprint 5	Hasil pengembangan halaman detail pesanan dan riwayat pesanan dibahas bersama supervisor dan anggota tim. Berdasarkan hasil diskusi, pengembangan halaman detail pesanan dan riwayat pesanan telah selesai sehingga tahap selanjutnya dapat dilanjutkan ke proses pengujian.
Sprint 6	Hasil pengujian menunjukkan bahwa seluruh fitur pada modul <i>customer</i> telah berjalan sesuai dengan kebutuhan sistem dan seluruh target pengembangan pada modul <i>customer</i> telah berhasil diselesaikan.

Berdasarkan hasil *Scrum meeting* pada setiap *sprint*, masukan dari supervisor dan anggota tim digunakan sebagai bahan evaluasi terhadap hasil pengembangan yang telah dilakukan. Masukan tersebut digunakan sebagai bahan evaluasi untuk memastikan setiap fitur telah sesuai dengan kebutuhan sistem sebelum pengembangan dilanjutkan ke *sprint* berikutnya. Apabila masih terdapat bagian yang perlu disesuaikan, penyempurnaan terlebih dahulu dilakukan berdasarkan hasil diskusi yang telah dilakukan. Dengan demikian, *Scrum meeting* membantu proses evaluasi hasil pengembangan serta menjadi acuan dalam menentukan pekerjaan pada *sprint* berikutnya.

3.2 Analisis dan Perancangan Sistem

Sebelum proses pengembangan dimulai, diperlukan tahap analisis dan perancangan untuk memperoleh gambaran yang jelas mengenai kebutuhan sistem yang akan dibangun. Tahap ini membantu tim memahami fungsi yang diperlukan, alur penggunaan sistem, kebutuhan data, serta hubungan antar komponen yang terlibat dalam proses pemesanan. Kegiatan analisis dilakukan dengan mempelajari kebutuhan pengguna serta proses bisnis yang diterapkan pada sistem CAVAA. Informasi yang diperoleh kemudian digunakan sebagai dasar dalam menyusun rancangan sistem yang akan dikembangkan. Melalui tahapan ini, proses implementasi dapat dilakukan dengan lebih terarah karena seluruh kebutuhan telah diidentifikasi sejak awal.

Hasil dari analisis dan perancangan mencakup kebutuhan sistem, gambaran interaksi pengguna dengan sistem, alur aktivitas yang terjadi selama proses pemesanan, struktur basis data, serta rancangan arsitektur sistem. Seluruh hasil tersebut menjadi acuan selama proses pengembangan fitur pada modul *customer*.

3.2.1 Analisis Kebutuhan Sistem

Analisis kebutuhan sistem dilakukan untuk mengidentifikasi layanan yang perlu tersedia pada modul *customer* serta kebutuhan pendukung yang diperlukan agar sistem dapat digunakan dengan baik. Identifikasi dilakukan berdasarkan proses bisnis yang diterapkan pada sistem CAVAA, kebutuhan pengguna, dan hasil diskusi yang dilakukan bersama tim pengembang selama proses pengembangan.

Kebutuhan yang berhasil diidentifikasi kemudian dibagi menjadi dua kelompok, yaitu kebutuhan fungsional dan kebutuhan nonfungsional. Kebutuhan fungsional berkaitan dengan layanan yang harus disediakan oleh sistem kepada pengguna. Sementara itu, kebutuhan nonfungsional berhubungan dengan kualitas dan karakteristik sistem yang perlu dipenuhi agar proses penggunaan dapat berjalan dengan baik. Daftar kebutuhan fungsional dan kebutuhan nonfungsional yang digunakan sebagai dasar pengembangan sistem dapat dilihat pada Tabel 3.3 dan Tabel 3.4.

Tabel 3.3 Kebutuhan Fungsional Sistem

No	Kebutuhan Fungsional
1	<i>Customer</i> dapat melakukan registrasi akun
2	<i>Customer</i> dapat melakukan <i>login</i>
3	<i>Customer</i> dapat mengakses halaman pemesanan melalui <i>QR Code</i>
4	Sistem dapat mengatur akses pemesanan berdasarkan status meja
5	<i>Customer</i> dapat melihat daftar menu
6	<i>Customer</i> dapat melihat detail produk
7	<i>Customer</i> dapat menambahkan produk ke keranjang
8	<i>Customer</i> dapat mengubah jumlah pesanan
9	<i>Customer</i> dapat menghapus produk dari keranjang
10	<i>Customer</i> dapat melakukan <i>checkout</i>
11	<i>Customer</i> dapat melihat detail pesanan
12	<i>Customer</i> dapat melihat riwayat pesanan

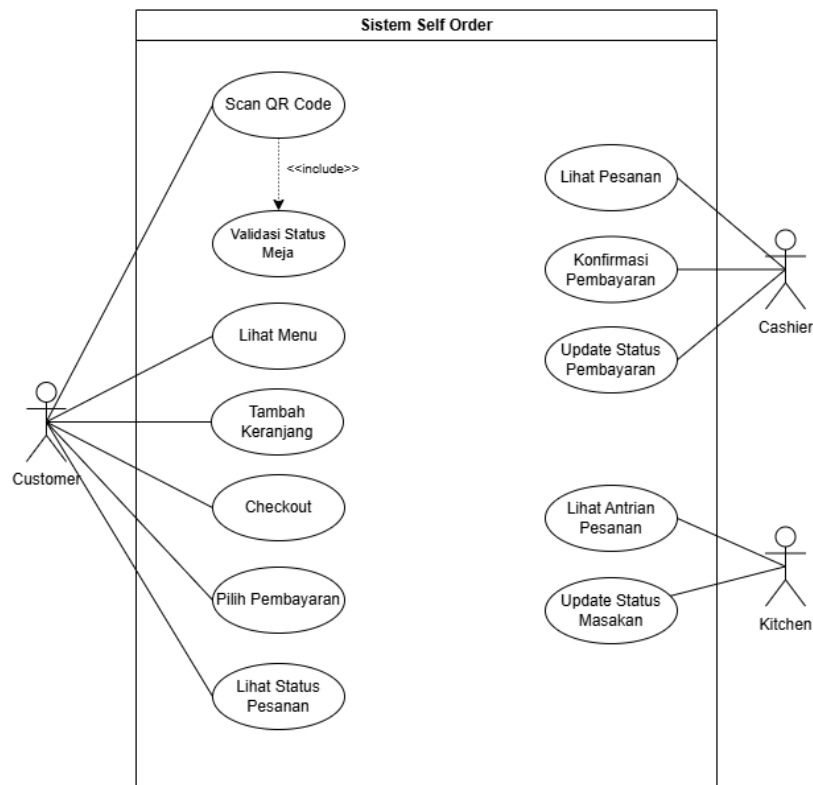
Tabel 3.4 Kebutuhan Nonfungsional Sistem

No	Kebutuhan Nonfungsional
1	Tampilan sistem dapat menyesuaikan berbagai ukuran layar perangkat
2	Sistem mampu menampilkan informasi dan memproses permintaan pengguna dengan baik
3	Data pengguna dan transaksi tersimpan pada basis data
4	Sistem memiliki antarmuka yang mudah dipahami oleh pengguna

3.2.2 Use Case Diagram

Use Case Diagram memberikan gambaran mengenai interaksi yang terjadi antara pengguna dan sistem pada proses *self-order* yang dikembangkan. Diagram ini membantu menunjukkan aktivitas yang dapat dilakukan oleh setiap pengguna sesuai peran yang dimiliki serta layanan yang disediakan oleh sistem.

Pada sistem CAVAA, terdapat beberapa pengguna yang terlibat dalam proses operasional, yaitu *customer*, *cashier*, dan *kitchen*. Masing-masing pengguna memiliki aktivitas yang berbeda sesuai kebutuhan pekerjaannya. *Customer* berinteraksi dengan sistem untuk melakukan pemesanan makanan dan minuman, *cashier* bertanggung jawab terhadap pengelolaan transaksi, sedangkan *kitchen* menangani pesanan yang masuk dari pelanggan. Melalui diagram ini, hubungan antara pengguna dan fungsi sistem dapat terlihat dengan lebih jelas. *Use Case Diagram* dapat dilihat pada Gambar 3.2.



Gambar 3.2 Use Case Diagram Sistem Self-Order

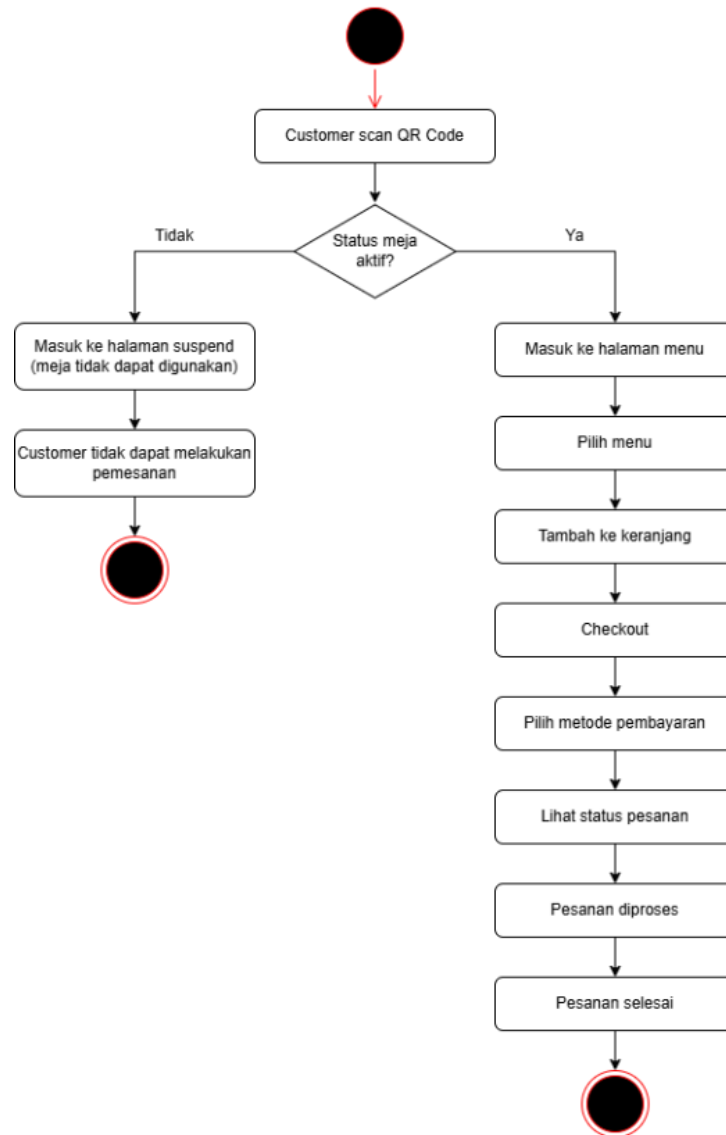
3.2.3 Activity Diagram

Activity Diagram menggambarkan urutan aktivitas yang berlangsung pada sistem selama pelanggan melakukan pemesanan. Diagram ini memperlihatkan alur proses mulai dari pelanggan mengakses sistem hingga pesanan berhasil dibuat. Dengan adanya diagram tersebut, setiap tahapan yang terjadi selama proses pemesanan dapat dipahami dengan lebih mudah.

Proses dimulai ketika pelanggan memindai *QR Code* yang tersedia pada meja restoran menggunakan *smartphone*. Setelah menerima informasi meja, sistem akan memeriksa status meja untuk memastikan bahwa meja tersebut dapat digunakan dalam proses pemesanan. Jika status meja tersedia, pelanggan akan diarahkan menuju halaman menu. Namun, apabila meja tidak dapat digunakan, sistem akan menampilkan informasi sesuai kondisi meja yang sedang berlangsung.

Setelah berhasil masuk ke halaman menu, pelanggan dapat melihat daftar produk yang tersedia dan memilih produk sesuai kebutuhan. Produk yang dipilih akan ditambahkan ke keranjang pesanan sebelum pelanggan melanjutkan ke proses *checkout*. Pada tahap ini, pelanggan dapat meninjau kembali pesanan yang telah dipilih sekaligus menentukan metode pembayaran yang akan digunakan.

Ketika proses *checkout* berhasil dilakukan, sistem akan membuat data pesanan dan menyimpan informasi transaksi ke dalam basis data. Selanjutnya, pelanggan dapat melihat informasi detail pesanan yang berisi daftar produk, status pesanan, serta informasi pembayaran yang berkaitan dengan transaksi tersebut. *Activity Diagram* sistem *self-order* dapat dilihat pada Gambar 3.3



Gambar 3.3 *Activity Diagram* proses pemesanan

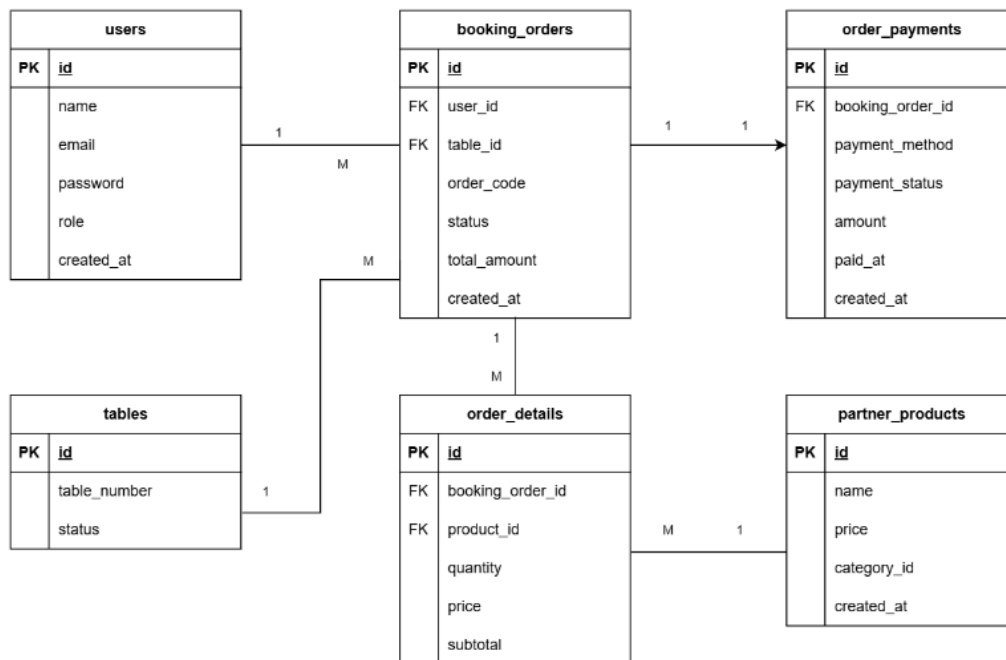
3.2.4 *Entity Relationship Diagram* (ERD)

Entity Relationship Diagram (ERD) digunakan untuk menunjukkan hubungan antar data yang digunakan pada sistem *self-order*. Melalui diagram ini, struktur basis data yang mendukung proses operasional sistem dapat dipahami dengan lebih jelas, termasuk hubungan

antara tabel yang saling berinteraksi selama proses pemesanan berlangsung. Beberapa tabel utama yang digunakan antara lain:

- users*: Menyimpan data pengguna
- tables*: Menyimpan data meja beserta status meja
- booking_orders*: Menyimpan data pesanan pelanggan
- order_details*: Menyimpan detail dari produk
- partner_products*: Menyimpan data produk
- order_payments*: Menyimpan data pembayaran

Relasi antar tabel digunakan untuk menghubungkan data pengguna, data pesanan, detail produk, data pembayaran, serta pengelolaan status meja sehingga seluruh proses dapat berjalan secara terintegrasi. *Entity Relationship Diagram* sistem dapat dilihat pada Gambar 3.4.



Gambar 3.4 *Entity Relationship Diagram* Sistem *Self-Order*

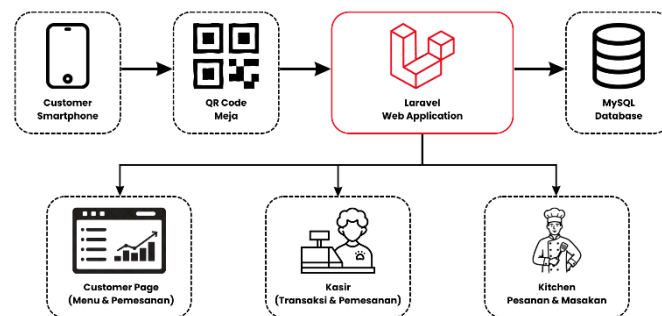
3.2.5 Arsitektur Sistem

Arsitektur sistem memberikan gambaran mengenai cara setiap komponen saling berinteraksi dalam mendukung proses *self-order* yang diterapkan pada CAVAA. Melalui rancangan ini, alur pertukaran data antara pengguna, aplikasi, basis data, dan layanan pendukung dapat dipahami secara lebih jelas.

Pelanggan mengakses sistem menggunakan *smartphone* melalui *QR Code* yang tersedia pada meja restoran. Setelah *QR Code* dipindai, pelanggan akan diarahkan ke modul *customer*

yang berjalan pada aplikasi berbasis web. Setiap permintaan yang dilakukan pelanggan diproses oleh server dan diteruskan ke basis data sesuai kebutuhan sistem. Basis data berperan sebagai tempat penyimpanan berbagai informasi yang berkaitan dengan pengguna, produk, meja, pesanan, dan pembayaran. Data yang tersimpan tersebut kemudian dapat diakses oleh modul lain yang terhubung dalam sistem, seperti *kitchen* dan *cashier*. Dengan mekanisme tersebut, informasi pesanan yang dibuat pelanggan dapat langsung diterima oleh bagian yang membutuhkan tanpa perlu dilakukan pencatatan ulang.

Pada proses pembayaran digital, sistem terhubung dengan layanan *payment gateway* untuk menangani transaksi yang dilakukan pelanggan. Informasi status pembayaran yang diterima dari layanan tersebut akan diperbarui secara otomatis pada sistem sehingga data transaksi dapat dipantau dengan lebih mudah. Melalui integrasi antar komponen tersebut, proses pemesanan, pengelolaan pesanan, dan pembayaran dapat berjalan dalam satu alur yang saling terhubung. Arsitektur sistem *self-order* dapat dilihat pada Gambar 3.5.



Gambar 3.5 Arsitektur sistem *self-order*

3.3 Implementasi Fitur

Pada tahap ini, hasil analisis kebutuhan dan perancangan sistem yang telah disusun sebelumnya mulai diterapkan ke dalam bentuk aplikasi web. Proses implementasi dilakukan pada modul *customer* dengan mengacu pada alur pemesanan yang diterapkan pada sistem CAVAA sehingga setiap fitur dapat mendukung proses transaksi yang berlangsung pada usaha kuliner. Fitur yang diimplementasikan pada modul *customer* meliputi autentikasi pengguna, penerapan *QR Code* dan manajemen sesi meja, halaman menu, keranjang pesanan, *checkout*, serta halaman detail dan riwayat pesanan. Setiap fitur dikembangkan secara bertahap dan melalui proses pengujian selama pengembangan untuk memastikan seluruh fungsi dapat berjalan sesuai dengan kebutuhan sistem.

3.3.1 Teknologi yang Digunakan

Dalam proses implementasi, digunakan beberapa teknologi yang mendukung pengembangan sistem pada sisi *backend*, *frontend*, basis data, serta pengelolaan kode sumber. Pemilihan teknologi dilakukan dengan menyesuaikan kebutuhan sistem dan standar pengembangan yang diterapkan pada proyek CAVAA. Teknologi yang digunakan selama proses pengembangan dapat dilihat pada Tabel 3.5.

Tabel 3.5 Teknologi yang Digunakan dalam Pengembangan Sistem

No	Teknologi	Fungsi
1	PHP	Digunakan sebagai bahasa pemrograman untuk mengembangkan logika bisnis dan berbagai proses yang berjalan pada sisi server
2	Laravel	Digunakan sebagai <i>framework</i> dalam pengembangan aplikasi web serta membantu pengelolaan alur kerja pada sisi <i>backend</i>
3	MySQL	Digunakan untuk menyimpan dan mengelola data pengguna, produk, meja, pesanan, dan transaksi yang terdapat pada sistem
4	JavaScript	Digunakan untuk menangani interaksi pengguna serta mendukung fitur yang memerlukan pembaruan data secara dinamis pada halaman web
5	Bootstrap	Digunakan untuk membantu pengembangan antarmuka yang responsif sehingga tampilan sistem dapat menyesuaikan berbagai ukuran perangkat
6	Visual Studio Code	Digunakan sebagai editor kode untuk membantu proses pengembangan, pengelolaan, dan pemeliharaan <i>source code</i> selama implementasi sistem
7	Git	Digunakan sebagai sistem pengelolaan versi kode sumber untuk mencatat perubahan program, mendukung kolaborasi antar anggota tim, serta memudahkan proses pengembangan dan pemeliharaan

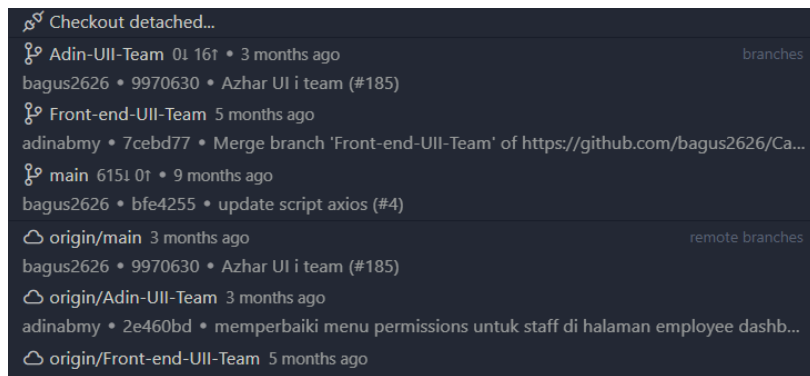
Berdasarkan Tabel 3.5, teknologi yang digunakan saling mendukung dalam proses pengembangan modul *customer* pada sistem CAVAA. Pemanfaatan teknologi tersebut membantu pengelolaan data, pengembangan antarmuka, implementasi logika bisnis, serta pengelolaan kode sumber selama proses pengembangan berlangsung.

3.3.2 Persiapan Lingkungan Pengembangan

Sebelum proses implementasi fitur dilakukan, lingkungan pengembangan terlebih dahulu disiapkan untuk mendukung pengerjaan modul *customer*. Tahap ini dilakukan untuk

memastikan sistem dapat dijalankan dengan baik sehingga proses pengembangan dapat berlangsung sesuai kebutuhan proyek.

Persiapan diawali dengan memperoleh *source code* proyek CAVAA yang digunakan oleh tim pengembang. Setelah *source code* berhasil diperoleh, pengembangan dilakukan menggunakan *branch* yang telah disediakan. Penggunaan *branch* bertujuan agar perubahan yang dilakukan tidak langsung memengaruhi *branch* utama yang digunakan pada proyek. Daftar *branch* yang tersedia selama proses pengembangan dapat dilihat pada Gambar 3.6.



Gambar 3.6 *Branch* Pengembangan Proyek CAVAA

Berdasarkan Gambar 3.6, proyek CAVAA menggunakan beberapa *branch* untuk mendukung proses pengembangan yang dilakukan oleh anggota tim. Melalui mekanisme tersebut, setiap anggota tim dapat mengerjakan tugas yang menjadi tanggung jawabnya masing-masing tanpa mengganggu pekerjaan anggota tim lainnya. Perubahan yang telah selesai dikerjakan kemudian dapat ditinjau terlebih dahulu sebelum digabungkan ke dalam *branch* utama proyek.

Setelah proses tersebut, *source code* proyek dibuka menggunakan Visual Studio Code sebagai lingkungan pengembangan utama. Melalui perangkat lunak tersebut, seluruh komponen pada proyek dapat diakses serta perubahan kode dapat dilakukan sesuai kebutuhan pengembangan. Struktur proyek yang digunakan selama kegiatan pengembangan dapat dilihat pada Gambar 3.7.



Gambar 3.7 Struktur Proyek CAVAA

Berdasarkan Gambar 3.7, proyek CAVAA tersusun ke dalam beberapa direktori yang digunakan untuk mendukung berbagai kebutuhan sistem. Selama kegiatan magang, proses pengembangan lebih banyak memanfaatkan direktori *app*, *resources*, dan *routes* karena bagian tersebut berkaitan langsung dengan pengembangan modul *customer*. Direktori *app* digunakan untuk mengelola logika sistem, direktori *resources* digunakan untuk menyimpan tampilan yang ditampilkan kepada pengguna, sedangkan direktori *routes* digunakan untuk mengatur alur akses halaman pada sistem.

Setelah proyek berhasil dibuka, konfigurasi sistem dilakukan melalui file *.env* yang digunakan untuk mengatur berbagai kebutuhan pengembangan, termasuk koneksi basis data yang digunakan oleh sistem. Konfigurasi ini diperlukan agar sistem dapat dijalankan sesuai dengan lingkungan kerja yang digunakan selama proses pengembangan. Tampilan konfigurasi sistem dapat dilihat pada Gambar 3.8.

```

APP_NAME=CAVAA
APP_ENV=local

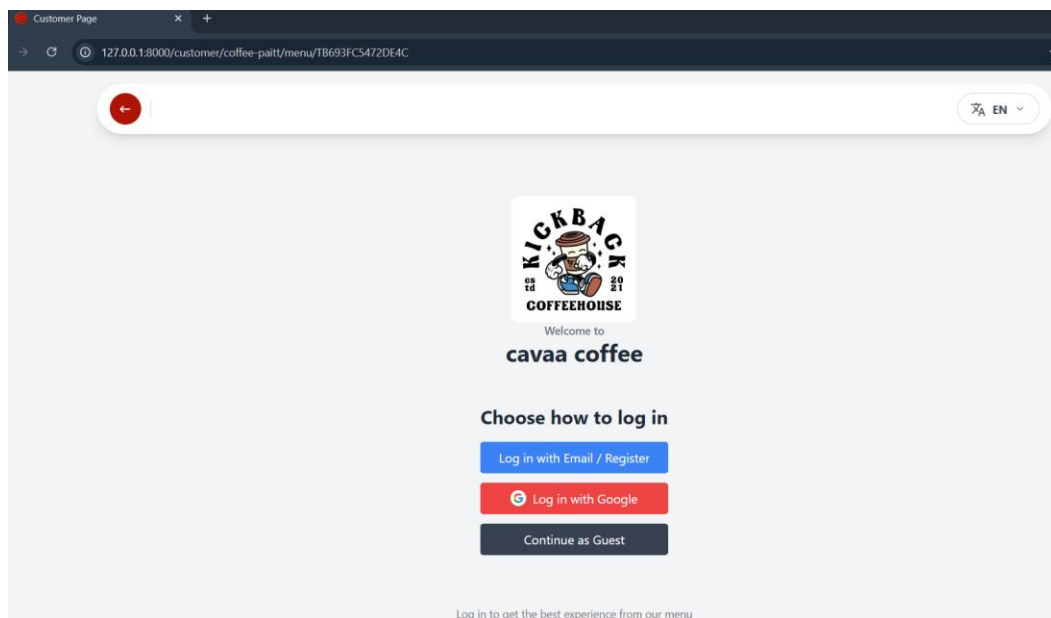
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=*****
DB_USERNAME=*****
DB_PASSWORD=*****

```

Gambar 3.8 Konfigurasi Sistem pada File .env

Melalui konfigurasi tersebut, sistem dapat terhubung dengan basis data serta layanan lain yang diperlukan selama proses pengembangan. Penyesuaian dilakukan sesuai kebutuhan proyek sehingga seluruh komponen dapat berfungsi dengan baik ketika sistem dijalankan.

Tahap berikutnya adalah memastikan sistem dapat berjalan dengan baik setelah seluruh konfigurasi selesai dilakukan. Proses ini dilakukan untuk memeriksa apakah seluruh komponen yang dibutuhkan telah terhubung dan dapat digunakan sebagaimana mestinya sebelum implementasi fitur dimulai. Tampilan sistem yang berhasil dijalankan dapat dilihat pada Gambar 3.9.



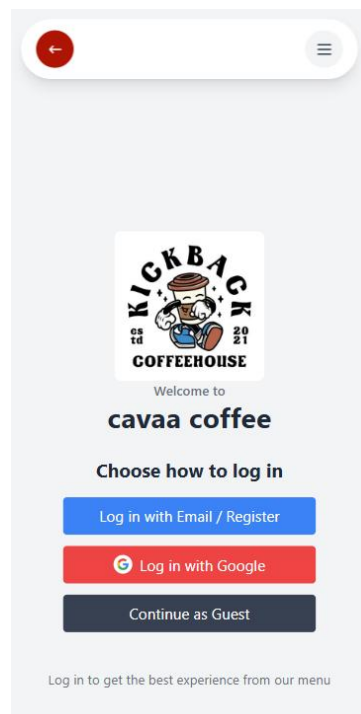
Gambar 3.9 Sistem CAVAA Berhasil Dijalankan

Setelah sistem berhasil dijalankan, proses pengembangan fitur pada modul *customer* dapat dilakukan. Dengan seluruh kebutuhan yang telah tersedia, proses implementasi fitur dapat dilanjutkan sesuai target pengembangan yang telah ditentukan.

3.3.3 Autentikasi *Customer*

Sebelum menggunakan layanan *self-order*, pelanggan perlu melakukan autentikasi terlebih dahulu. Saat halaman pemesanan diakses, sistem akan menampilkan beberapa pilihan *login* yang dapat digunakan pelanggan untuk masuk ke dalam sistem. Melalui proses ini, aktivitas pemesanan yang dilakukan pelanggan dapat terhubung dengan akun yang digunakan sehingga data pesanan dapat tersimpan dengan sesuai.

Ketika pelanggan mengakses halaman pemesanan, sistem akan menampilkan halaman pilihan *login*. Pada halaman tersebut tersedia beberapa opsi yang dapat dipilih pelanggan untuk masuk ke dalam sistem.



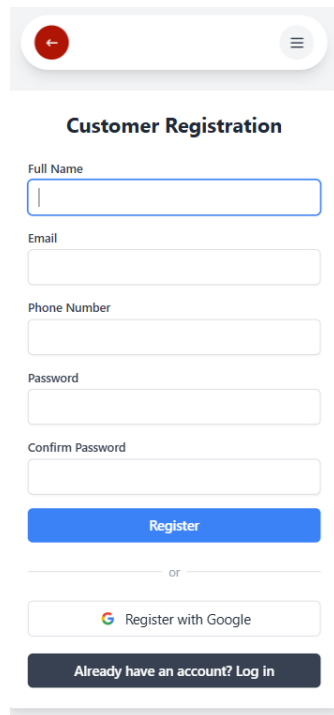
Gambar 3.10 Halaman Pilihan *Login Customer*

Berdasarkan Gambar 3.10, pelanggan dapat masuk menggunakan akun yang telah terdaftar, akun Google, maupun melanjutkan penggunaan sistem sebagai *guest*. Penyediaan beberapa metode *login* tersebut memberikan kemudahan bagi pelanggan untuk memilih cara akses sesuai dengan kebutuhan masing-masing.

Apabila pelanggan memilih masuk menggunakan akun yang telah terdaftar, sistem akan menampilkan halaman *login* seperti yang ditunjukkan pada Gambar 3.11.

Gambar 3.11 Halaman *Login Customer*

Pada halaman *login*, pelanggan diminta memasukkan alamat email dan *password* yang telah didaftarkan sebelumnya. Data yang dimasukkan akan diperiksa terlebih dahulu sebelum sistem memberikan akses ke halaman menu. Selain melakukan validasi akun, sistem juga memeriksa status verifikasi email pelanggan. Jika email belum terverifikasi, sistem akan mengirim ulang tautan verifikasi dan mengarahkan pelanggan ke halaman verifikasi sebelum pelanggan dapat menggunakan layanan yang tersedia. Bagi pelanggan yang belum memiliki akun, sistem menyediakan halaman registrasi yang dapat digunakan untuk membuat akun baru. Tampilan halaman registrasi dapat dilihat pada Gambar 3.12.

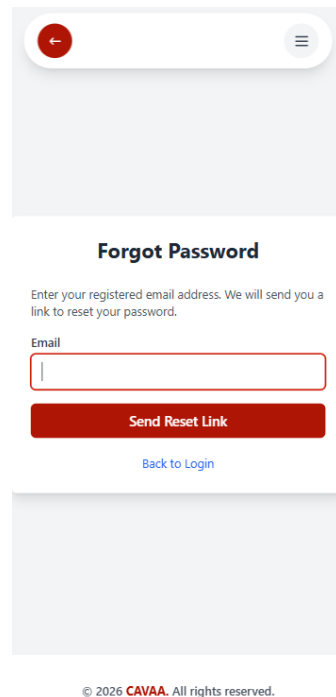


The image shows a mobile application interface for customer registration. At the top, there is a navigation bar with a red circular button on the left and a hamburger menu icon on the right. Below the navigation bar, the title "Customer Registration" is centered. The form consists of several input fields: "Full Name", "Email", "Phone Number", "Password", and "Confirm Password". Each field is a simple white rectangle with a light gray border. Below the "Confirm Password" field is a blue button with the text "Register" in white. Underneath the button is a horizontal line with the word "or" centered. Below the line is a button with the Google logo and the text "Register with Google". At the bottom is a dark gray button with the text "Already have an account? Log in" in white.

Gambar 3.12 Halaman Registrasi *Customer*

Pada halaman registrasi, pelanggan dapat mengisi data yang diperlukan untuk membuat akun baru, seperti nama lengkap, alamat email, nomor telepon, dan *password*. Data yang berhasil didaftarkan kemudian disimpan ke dalam basis data dan digunakan sebagai identitas pelanggan saat menggunakan layanan pada sistem. Setelah proses registrasi selesai dilakukan, sistem akan mengirimkan email verifikasi untuk memastikan bahwa alamat email yang digunakan telah valid sebelum akun dapat digunakan sepenuhnya.

Selain menyediakan fitur registrasi dan *login*, sistem juga dilengkapi fitur lupa *password* untuk membantu pelanggan yang mengalami kendala saat mengakses akun. Tampilan halaman lupa *password* dapat dilihat pada Gambar 3.13.



Gambar 3.13 Halaman Lupa *Password*

Melalui halaman tersebut, pelanggan dapat memasukkan alamat email yang telah terdaftar pada sistem untuk menerima tautan reset *password*. Setelah tautan dikirimkan, pelanggan dapat membuat *password* baru dan kembali menggunakan akun yang dimiliki.

Pada proses reset *password*, sistem menyimpan informasi *outlet* dan meja yang sedang digunakan ke dalam *session*. Penyimpanan informasi ini dilakukan agar pelanggan tetap dapat melanjutkan proses pemesanan pada *outlet* dan meja yang sama setelah reset *password* berhasil dilakukan. Sebelum tautan reset *password* dikirimkan, sistem terlebih dahulu menyimpan informasi tersebut ke dalam *session*. Potongan kode yang digunakan untuk menyimpan informasi tersebut dapat dilihat pada Gambar 3.14.

```
session([
  'customer.partner_slug' => $partner_slug,
  'customer.table_code'   => $table_code,
]);
```

Gambar 3.14 Penyimpanan Informasi *Outlet* dan Meja pada *Session*

Gambar 3.14 menunjukkan proses penyimpanan informasi *outlet* dan kode meja ke dalam *session*. Informasi tersebut digunakan selama proses reset *password* berlangsung sehingga

pelanggan tidak perlu melakukan pemindaian *QR Code* kembali setelah *password* berhasil diperbarui.

Setelah *password* berhasil diperbarui, pelanggan tidak perlu melakukan *login* secara manual. Sistem akan melakukan autentikasi secara otomatis terhadap akun yang telah menyelesaikan proses reset *password*. Potongan kode yang digunakan untuk melakukan proses tersebut dapat dilihat pada Gambar 3.15.

```
Auth::guard('customer')->login($customer);
```

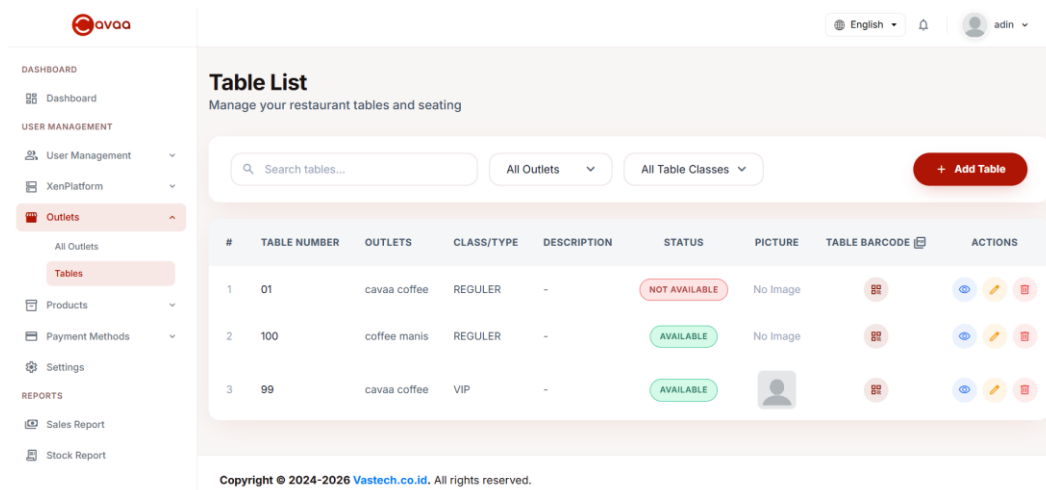
Gambar 3.15 *Login Otomatis Setelah Reset Password*

Gambar 3.15 menunjukkan proses *login* otomatis menggunakan *guard customer* setelah *password* berhasil diperbarui. Melalui mekanisme ini, pelanggan dapat langsung kembali menggunakan layanan yang tersedia tanpa harus mengulangi proses *login* dari awal.

3.3.4 Implementasi *QR Code* dan Manajemen Sesi Meja

Sebelum pelanggan dapat melakukan pemesanan, sistem terlebih dahulu memanfaatkan *QR Code* untuk menghubungkan pelanggan dengan meja yang sedang digunakan. Setiap meja yang tersedia pada *outlet* memiliki *QR Code* yang berbeda sehingga sistem dapat mengenali lokasi pelanggan saat proses pemesanan berlangsung. Sistem juga menerapkan pengelolaan status meja untuk memastikan bahwa transaksi hanya dapat dilakukan pada meja yang masih tersedia.

Sebelum *QR Code* dapat digunakan, pemilik usaha perlu menambahkan data meja ke dalam sistem. Setiap meja yang berhasil ditambahkan akan memperoleh kode unik sebagai identitas meja. Berdasarkan kode tersebut, sistem akan membuat *QR Code* secara otomatis yang nantinya digunakan sebagai akses menuju halaman pemesanan *customer*. Tampilan halaman pengelolaan *QR Code* meja dapat dilihat pada Gambar 3.16.



Gambar 3.16 Halaman Pengelolaan *QR Code* Meja

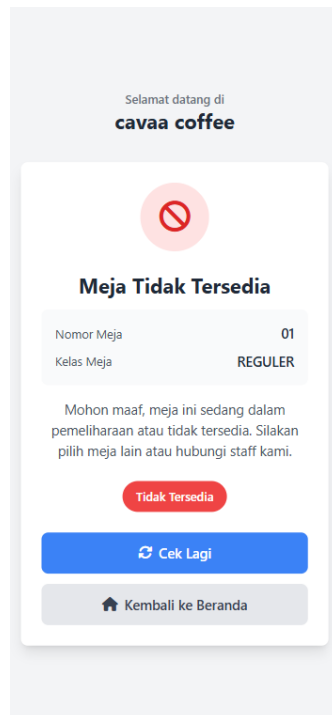
Pada halaman tersebut, sistem menampilkan daftar meja yang telah terdaftar beserta *QR Code* untuk masing-masing meja. *QR Code* dapat diunduh maupun dicetak untuk ditempatkan pada meja yang sesuai. Dengan cara ini, pelanggan cukup melakukan pemindaian menggunakan *smartphone* untuk mengakses halaman pemesanan tanpa perlu memasukkan informasi meja secara manual.

Setelah *QR Code* dipindai, sistem akan menerima informasi berupa identitas *outlet* dan kode meja yang digunakan pelanggan. Informasi tersebut kemudian digunakan untuk mencari data meja pada basis data. Setelah data meja ditemukan, sistem akan memeriksa status meja terlebih dahulu sebelum pelanggan diarahkan ke halaman pemesanan. Proses validasi tersebut dapat dilihat pada Gambar 3.17.

```
if ($table->status === 'available') {
    return redirect()->route('customer.menu.index', [
        'partner_slug' => $partner_slug,
        'table_code' => $table_code
    ]);
}
```

Gambar 3.17 Validasi Status Meja

Pada proses ini, sistem memeriksa status yang dimiliki oleh meja sebelum pelanggan diarahkan ke halaman menu. Jika status meja tersedia (*available*), pelanggan dapat melanjutkan proses pemesanan. Sebaliknya, jika status meja tidak tersedia (*not available*), pelanggan tidak dapat mengakses halaman pemesanan dan sistem akan menampilkan informasi mengenai kondisi meja tersebut. Ketika meja berada pada kondisi yang tidak dapat digunakan, sistem akan menampilkan halaman informasi kepada pelanggan. Tampilan halaman tersebut dapat dilihat pada Gambar 3.18.

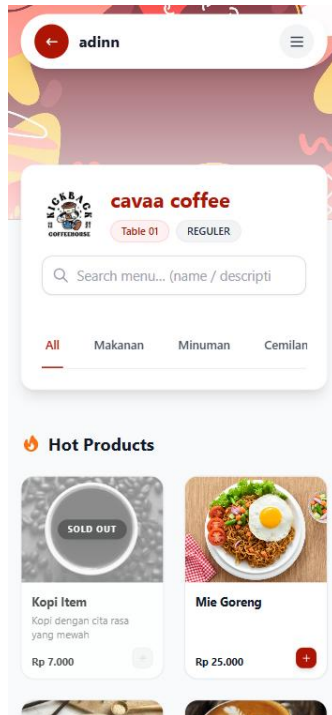


Gambar 3.18 Halaman Status Meja Tidak Tersedia

Pada halaman tersebut, sistem memberikan informasi bahwa meja yang dipilih sedang dalam proses pemeliharaan sehingga belum dapat digunakan untuk melakukan pemesanan. Informasi ini membantu pelanggan mengetahui kondisi meja yang sedang digunakan dan memastikan transaksi hanya dilakukan pada meja yang tersedia.

3.3.5 Halaman *Customer*

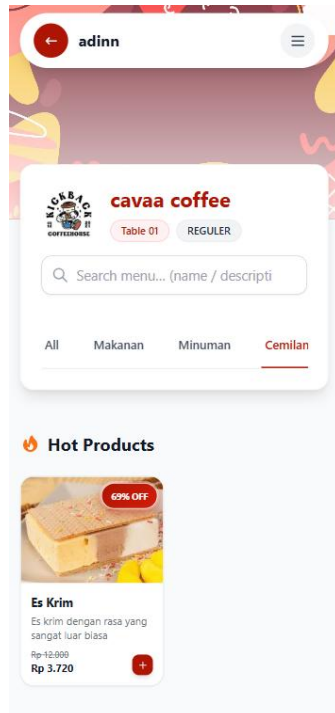
Setelah proses autentikasi dan validasi status meja berhasil dilakukan, pelanggan akan diarahkan ke halaman *customer*. Pada halaman ini, sistem menampilkan daftar produk yang tersedia sesuai dengan *outlet* yang sedang diakses sehingga pelanggan dapat memilih menu yang ingin dipesan. Tampilan halaman menu *customer* dapat dilihat pada Gambar 3.19.



Gambar 3.19 Halaman Menu *Customer*

Pada halaman *customer*, sistem menampilkan informasi produk seperti nama menu, gambar produk, harga, dan deskripsi singkat. Informasi tersebut membantu pelanggan mengenali produk yang tersedia sebelum melakukan pemesanan. Sistem juga menampilkan produk yang sedang memiliki promosi sehingga pelanggan dapat langsung melihat harga yang berlaku.

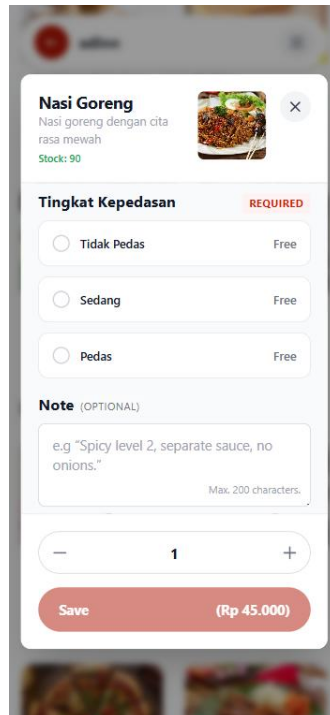
Untuk memudahkan proses pencarian menu, sistem menyediakan fitur pengelompokan produk berdasarkan kategori. Tampilan kategori produk dapat dilihat pada Gambar 3.20.



Gambar 3.20 Kategori Produk

Pengelompokan produk berdasarkan kategori membantu pelanggan dalam menemukan menu yang diinginkan tanpa harus menelusuri seluruh daftar produk yang tersedia. Fitur ini menjadi penting terutama ketika jumlah menu yang dimiliki oleh *outlet* cukup banyak.

Ketika pelanggan memilih salah satu produk, sistem akan menampilkan halaman detail produk yang berisi informasi yang lebih lengkap. Tampilan halaman detail produk dapat dilihat pada Gambar 3.21.



Gambar 3.21 Halaman Detail Produk

Pada halaman detail produk, pelanggan dapat melihat informasi produk secara lebih rinci, termasuk harga, deskripsi, pilihan varian, serta opsi tambahan yang tersedia. Halaman ini juga digunakan untuk menentukan jumlah pesanan sebelum produk dimasukkan ke dalam keranjang.

Untuk menampilkan daftar menu pada halaman *customer*, sistem terlebih dahulu mengambil data produk yang dimiliki oleh *outlet* yang sedang diakses. Proses tersebut ditunjukkan pada Gambar 3.22.

```
$partner_products = PartnerProduct::with([
    'category',
    'parent_options.options',
    'promotion' => function ($q) {
        $q->activeToday();
    },
    'stock',
    'parent_options.options.stock',
])
->where('partner_id', $partner->id)
->where('is_active', 1)
->get();
```

Gambar 3.22 Pengambilan Data Produk

Gambar 3.22 menunjukkan proses pengambilan data produk yang akan ditampilkan pada halaman *customer*. Sistem mengambil seluruh produk aktif yang dimiliki oleh *outlet* berdasarkan identitas *outlet* yang sedang diakses pelanggan. Selain data produk utama, sistem

juga memuat informasi kategori, promosi yang sedang berlangsung, pilihan tambahan produk, serta data stok yang diperlukan selama proses pemesanan. Dengan cara tersebut, seluruh informasi yang dibutuhkan dapat ditampilkan secara bersamaan pada halaman menu sehingga proses akses data menjadi lebih efisien.

Halaman *customer* juga dilengkapi dengan mekanisme penyimpanan data keranjang pesanan. Fitur ini dikembangkan untuk menjaga agar produk yang telah dipilih pelanggan tidak langsung hilang ketika halaman dimuat ulang atau pelanggan berpindah ke halaman lain selama proses pemesanan masih berlangsung. Dengan mekanisme tersebut, pelanggan dapat melanjutkan proses pemesanan tanpa harus menambahkan kembali produk yang sebelumnya telah dipilih.

Data keranjang disimpan sementara pada *browser* menggunakan penyimpanan lokal (*local storage*). Setiap kali pelanggan menambahkan atau mengubah jumlah produk pada keranjang, sistem akan memperbarui data yang tersimpan beserta waktu penyimpanannya. Ketika halaman *customer* kembali diakses, sistem dapat menampilkan kembali produk yang sebelumnya telah dimasukkan ke dalam keranjang sehingga pelanggan tidak perlu mengulangi proses pemilihan menu dari awal. Mekanisme penyimpanan dan pengambilan kembali data keranjang tersebut dapat dilihat pada Gambar 3.23.

```
const CART_EXPIRY_MINUTES = {{ config('session.lifetime', 120) }};

function saveCartToStorage() {
  try {
    const cartData = {
      items: cart,
      lastKeyPerProduct: lastKeyPerProduct,
      timestamp: Date.now()
    };
    localStorage.setItem('menuCart', JSON.stringify(cartData));
  } catch (error) {
    console.error('Error saving cart:', error);
  }
}

function loadCartFromStorage() {
  try {
    const stored = localStorage.getItem('menuCart');
    if (!stored) return null;

    const cartData = JSON.parse(stored);
    const now = Date.now();
    const expiry = CART_EXPIRY_MINUTES * 60 * 1000;

    if (now - cartData.timestamp > expiry) {
      localStorage.removeItem('menuCart');
      return null;
    }
  }
}
```

```

        return cartData;
    } catch (error) {
        console.error('Error loading cart:', error);
        localStorage.removeItem('menuCart');
        return null;
    }
}

function clearCartFromStorage() {
    try {
        localStorage.removeItem('menuCart');
    } catch (error) {
        console.error('Error clearing cart:', error);
    }
}

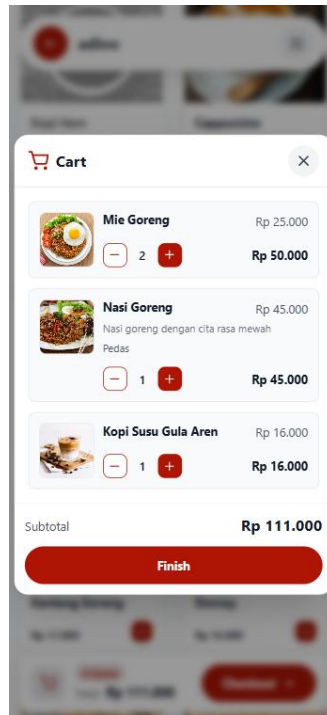
```

Gambar 3.23 Penyimpanan Data Keranjang pada *Browser*

Pada proses tersebut, sistem menyimpan informasi produk yang dipilih pelanggan beserta waktu penyimpanannya ke dalam *browser*. Saat halaman kembali diakses, sistem akan memuat kembali data yang tersimpan sehingga isi keranjang tetap tersedia selama sesi pemesanan masih berlangsung. Sistem juga melakukan pemeriksaan terhadap waktu penyimpanan yang telah ditentukan. Apabila batas waktu sesi telah terlewati atau proses *checkout* berhasil dilakukan, data keranjang akan dihapus secara otomatis sehingga tidak digunakan kembali pada transaksi berikutnya.

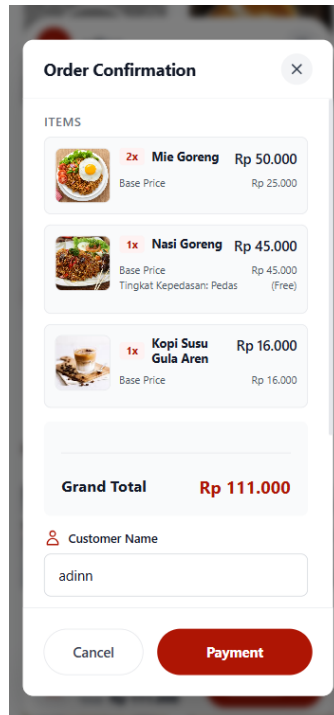
3.3.6 Keranjang Pesanan dan *Checkout*

Setelah memilih produk pada halaman menu, pelanggan dapat menambahkan produk tersebut ke dalam keranjang pesanan. Pada tahap ini, pelanggan dapat melihat kembali produk yang telah dipilih sebelum pesanan dikirimkan ke sistem. Informasi seperti jumlah pesanan, pilihan tambahan, dan produk yang dipilih dapat diperiksa kembali untuk memastikan bahwa pesanan yang dibuat sudah sesuai. Tampilan halaman keranjang pesanan dapat dilihat pada Gambar 3.24.



Gambar 3.24 Halaman Keranjang Pesanan

Pada halaman keranjang pesanan, sistem menampilkan daftar produk yang telah dipilih pelanggan beserta jumlah pesanan, harga produk, dan total biaya yang harus dibayarkan. Pelanggan juga dapat mengubah jumlah pesanan maupun menghapus produk tertentu dari keranjang sebelum melanjutkan ke tahap berikutnya. Setelah seluruh pesanan dipastikan sesuai, pelanggan dapat melanjutkan ke halaman *checkout*. Tampilan halaman *checkout* dapat dilihat pada Gambar 3.25.

Gambar 3.25 Halaman *Checkout*

Pada halaman *checkout*, sistem menampilkan ringkasan pesanan yang akan diproses beserta informasi meja yang sedang digunakan. Pelanggan juga dapat memilih metode pembayaran yang tersedia sebelum pesanan dikirimkan ke sistem. Data yang telah dikonfirmasi pada halaman ini selanjutnya digunakan dalam proses pembuatan transaksi pesanan. Sebelum transaksi dibuat, sistem melakukan pemeriksaan ulang terhadap ketersediaan stok produk yang terdapat pada keranjang pesanan. Pemeriksaan ini dilakukan untuk memastikan bahwa jumlah produk yang dipesan masih sesuai dengan kondisi stok terbaru pada sistem. Selama proses pemesanan berlangsung, ketersediaan stok dapat berubah karena adanya transaksi lain yang terjadi pada waktu yang hampir bersamaan. Oleh sebab itu, sistem melakukan validasi ulang sebelum pesanan disimpan agar jumlah produk yang dipesan tetap sesuai dengan stok yang tersedia saat transaksi dilakukan. Implementasi validasi stok dapat dilihat pada Gambar 3.26.

```

$available = ($productStock->quantity ?? 0) - ($productStock->quantity_reserved ?? 0);

if (!$productStock || $available < $qty) {
    throw new \Exception(
        "Stok {$product->name} (Produk) tidak mencukupi."
    );
}

```

Gambar 3.26 Validasi Ketersediaan Stok Saat *Checkout*

Gambar 3.26 menunjukkan proses pemeriksaan ketersediaan stok yang dilakukan sebelum transaksi dibuat. Pada tahap ini, sistem menghitung jumlah stok yang masih dapat digunakan dengan memperhitungkan stok yang telah direservasi oleh transaksi lain. Hasil perhitungan tersebut kemudian dibandingkan dengan jumlah produk yang dipesan pelanggan. Apabila stok yang tersedia tidak mencukupi, proses *checkout* akan dihentikan dan sistem menampilkan informasi kepada pelanggan mengenai ketersediaan stok produk. Mekanisme ini membantu mencegah terjadinya pemesanan melebihi stok yang tersedia sehingga jumlah produk yang dipesan tetap sesuai dengan kondisi stok pada saat transaksi dilakukan.

Setelah seluruh proses validasi berhasil dilakukan, sistem akan membuat data transaksi berdasarkan informasi pesanan yang telah dikirimkan pelanggan. Proses pembuatan data transaksi ditunjukkan pada Gambar 3.27.

```

$booking_order = BookingOrder::create([
    'booking_order_code' => $booking_order_code,
    'partner_id' => $partner->id,
    'partner_name' => $partner->name,
    'table_id' => $table->id,
    'customer_id' => $customer ? $customer->id : null,
    'order_by' => 'CUSTOMER',
    'customer_name' => $customer ? $customer->name : 'guest-' . $request->order_name,
    'order_status' => 'UNPAID',
    'payment_method' => $payment_method,
    'total_order_value' => $request->total_amount,
]);

```

Gambar 3.27 Pembuatan Data Transaksi Pesanan

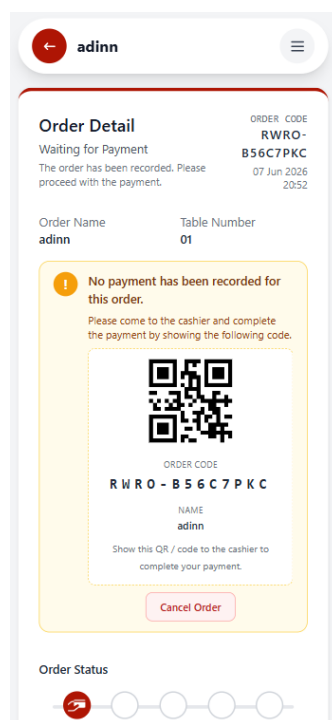
Gambar 3.27 menunjukkan proses pembuatan data transaksi ketika pelanggan menyelesaikan proses *checkout*. Sistem menyimpan berbagai informasi pesanan, seperti identitas pelanggan, meja yang digunakan, metode pembayaran yang dipilih, serta total nilai transaksi. Data tersebut kemudian digunakan sebagai acuan untuk proses pembayaran dan pengelolaan pesanan pada tahap berikutnya.

Setelah transaksi berhasil dibuat, sistem akan menyimpan detail produk yang dipesan beserta informasi tambahannya. Selanjutnya pesanan akan diteruskan ke bagian operasional untuk diproses sesuai dengan alur pelayanan yang berlaku pada restoran.

3.3.7 Halaman Detail Pesanan dan Riwayat Pesanan

Setelah proses *checkout* berhasil dilakukan, pelanggan dapat melihat informasi transaksi yang telah dibuat melalui halaman detail pesanan dan riwayat pesanan. Pada halaman ini, sistem menampilkan informasi mengenai status pesanan, rincian produk yang dipesan, informasi pembayaran, serta daftar transaksi yang pernah dilakukan sebelumnya. Informasi tersebut membantu pelanggan memantau kondisi pesanan secara langsung tanpa perlu menanyakan status transaksi kepada pihak restoran.

Setelah transaksi berhasil dibuat, pelanggan akan diarahkan menuju halaman detail pesanan. Halaman ini menampilkan informasi utama yang berkaitan dengan transaksi yang sedang berlangsung. Tampilan halaman detail pesanan dapat dilihat pada Gambar 3.28.



Gambar 3.28 Informasi Transaksi dan *QR Code* Pembayaran

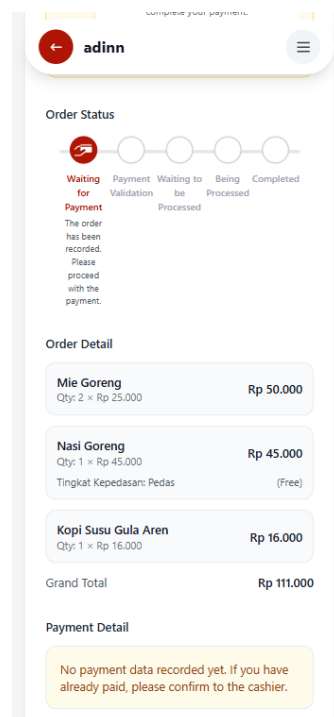
Pada halaman tersebut, sistem menampilkan informasi transaksi seperti kode pesanan, nama pelanggan, nomor meja, metode pembayaran, status pembayaran, serta status pesanan yang sedang berlangsung. Pelanggan juga dapat melihat *QR Code* transaksi yang digunakan sebagai identitas pesanan dan dapat ditunjukkan kepada kasir saat proses pembayaran.

Informasi tersebut membantu pelanggan mengetahui kondisi transaksi setelah pesanan berhasil dibuat. Untuk menampilkan *QR Code* transaksi, sistem terlebih dahulu memeriksa ketersediaan data *QR Code* yang telah dihasilkan. Potongan kode yang digunakan untuk menampilkan *QR Code* tersebut dapat dilihat pada Gambar 3.29.

```
@if(!empty($qrPngBase64))
    
@endif
```

Gambar 3.29 *QR Code* Transaksi

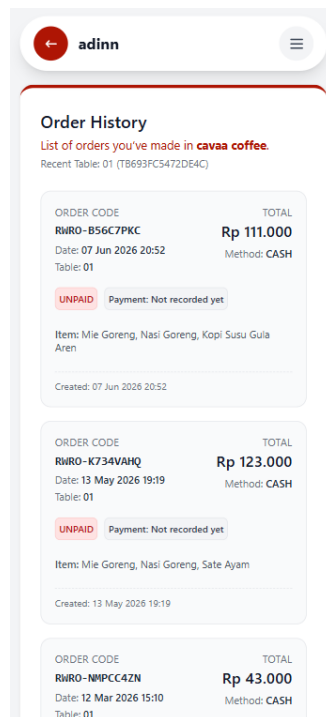
Potongan kode di atas digunakan untuk menampilkan *QR Code* transaksi pada halaman detail pesanan. *QR Code* hanya akan ditampilkan apabila datanya tersedia. *QR Code* tersebut dapat ditunjukkan kepada kasir saat proses pembayaran sehingga proses transaksi dapat dilakukan dengan lebih mudah. Halaman detail pesanan juga menampilkan rincian produk yang telah dipesan oleh pelanggan. Tampilan rincian pesanan dapat dilihat pada Gambar 3.30.



Gambar 3.30 Detail Produk Pesanan

Pada bagian ini sistem menampilkan daftar produk yang dipesan beserta jumlah pesanan, harga produk, subtotal, dan total pembayaran. Status pesanan dan informasi pembayaran juga dapat dilihat pada halaman ini sehingga seluruh informasi transaksi dapat dilihat secara lengkap. Pelanggan juga dapat mengakses halaman riwayat pesanan untuk melihat transaksi

yang pernah dilakukan sebelumnya. Tampilan halaman riwayat pesanan dapat dilihat pada Gambar 3.31.



Gambar 3.31 Halaman Riwayat Pesanan

Pada halaman tersebut, sistem menampilkan daftar transaksi yang pernah dilakukan pelanggan pada *outlet* yang sedang digunakan. Informasi yang ditampilkan meliputi kode pesanan, tanggal transaksi, total pembayaran, metode pembayaran, status pesanan, dan status pembayaran. Setiap transaksi ditampilkan dalam bentuk kartu sehingga informasi dapat dibaca dengan lebih mudah. Apabila jumlah riwayat transaksi cukup banyak, pelanggan dapat menampilkan data lainnya melalui tombol *load more*. Fitur ini membantu menjaga tampilan halaman, meskipun jumlah transaksi yang tersimpan terus bertambah. Potongan kode yang digunakan untuk mengambil data tambahan dapat dilihat pada Gambar 3.32.

```
fetch(nextUrl, {
  headers: {
    'X-Requested-With': 'XMLHttpRequest',
    'Accept': 'application/json'
  },
  credentials: 'same-origin'
})
```

Gambar 3.32 Pengambilan Data Riwayat Pesanan

Potongan kode pada Gambar 3.32 digunakan untuk mengambil data riwayat pesanan tambahan ketika pelanggan memilih tombol *load more*. Data yang diterima kemudian ditambahkan ke halaman tanpa perlu melakukan pemuatan ulang sehingga pelanggan dapat

menelusuri riwayat transaksi dengan lebih nyaman. Setiap kartu riwayat pesanan yang ditampilkan juga dapat dipilih untuk melihat informasi transaksi secara lebih jelas. Ketika salah satu riwayat pesanan dipilih, pelanggan akan diarahkan menuju halaman detail pesanan yang sesuai. Mekanisme tersebut ditunjukkan pada Gambar 3.33.

```
const url = card.getAttribute('data-detail-url');

if (url) {
    window.location.href = url;
}
```

Gambar 3.33 Navigasi ke Halaman Detail Pesanan

3.4 Pengujian Sistem

Setelah proses pengembangan pada modul *customer* selesai dilakukan, tahap selanjutnya adalah melakukan pengujian terhadap fitur-fitur yang telah dibuat. Pengujian dilakukan untuk memastikan bahwa setiap fitur dapat berjalan sesuai dengan kebutuhan sistem serta menghasilkan keluaran yang sesuai dengan proses yang dirancang. Pengujian juga digunakan untuk mengetahui pengalaman pengguna saat menggunakan fitur-fitur yang tersedia pada modul *customer*.

Pada modul *customer*, pengujian dilakukan dalam dua tahap, yaitu pengujian fungsional sistem dan pengujian *usability* menggunakan metode *System Usability Scale* (SUS). Pengujian fungsional digunakan untuk memeriksa kesesuaian fungsi pada setiap fitur yang dikembangkan, sedangkan pengujian *usability* dilakukan untuk memperoleh penilaian pengguna terhadap kemudahan penggunaan sistem.

3.4.1 Pengujian Fungsional Sistem

Pengujian fungsional dilakukan pada seluruh fitur utama yang terdapat pada modul *customer* untuk memastikan bahwa setiap fitur dapat digunakan sesuai dengan fungsinya. Pengujian ini difokuskan pada proses yang berkaitan dengan aktivitas pelanggan selama menggunakan layanan *self-order*, mulai dari autentikasi hingga melihat riwayat pesanan.

Proses pengujian dilakukan dengan menjalankan berbagai skenario penggunaan pada setiap fitur, kemudian membandingkan hasil yang diperoleh dengan hasil yang diharapkan. Skenario pengujian mencakup fitur autentikasi *customer*, pemindaian *QR Code*, manajemen sesi meja, halaman menu, keranjang pesanan, *checkout*, detail pesanan, dan riwayat pesanan. Hasil pengujian dapat dilihat pada Tabel 3.6.

Tabel 3.6 Hasil Pengujian Fungsional Sistem

No	Fitur yang Diuji	Skenario Pengujian	Hasil yang Diharapkan	Hasil
1	Autentikasi <i>Customer</i>	<i>Customer</i> melakukan <i>login</i> menggunakan akun yang terdaftar	<i>Customer</i> berhasil masuk ke sistem	Berhasil
2	Registrasi <i>Customer</i>	<i>Customer</i> mengisi data registrasi dengan benar	Akun <i>customer</i> berhasil dibuat	Berhasil
3	<i>Reset Password</i>	<i>Customer</i> melakukan proses lupa <i>password</i>	Mendapatkan <i>email confirmation</i> dan <i>password</i> berhasil diperbarui	Berhasil
4	<i>QR Code</i> Meja	<i>Customer</i> memindai <i>QR Code</i> meja	Sistem menampilkan halaman menu sesuai meja yang dipilih	Berhasil
5	Manajemen Sesi Meja	<i>Customer</i> mengakses meja yang tersedia	Sistem mengizinkan proses pemesanan	Berhasil
6	Halaman Menu	<i>Customer</i> membuka daftar menu	Daftar produk berhasil ditampilkan	Berhasil
7	Keranjang Pesanan	<i>Customer</i> menambahkan produk ke keranjang	Produk berhasil masuk ke keranjang	Berhasil
8	<i>Checkout</i>	<i>Customer</i> mengonfirmasi pesanan	Data transaksi berhasil dibuat	Berhasil
9	Detail Pesanan	<i>Customer</i> membuka detail transaksi	Informasi pesanan berhasil ditampilkan	Berhasil
10	Riwayat Pesanan	<i>Customer</i> membuka halaman riwayat pesanan	Riwayat transaksi berhasil ditampilkan	Berhasil

Berdasarkan hasil pengujian pada Tabel 3.6, seluruh skenario yang dijalankan berhasil menghasilkan keluaran yang sesuai dengan hasil yang diharapkan. Hasil tersebut menunjukkan bahwa fitur-fitur pada modul *customer* telah berfungsi dengan baik dan dapat digunakan untuk mendukung proses pemesanan secara mandiri.

3.4.2 Pengujian *Usability* Menggunakan *System Usability Scale* (SUS)

Pengujian juga dilakukan untuk mengetahui tingkat kemudahan penggunaan modul *customer* dari sudut pandang pengguna. Pengujian ini menggunakan metode *System Usability*

Scale (SUS) yang menghasilkan penilaian berdasarkan pengalaman pengguna setelah mencoba fitur-fitur yang tersedia pada sistem.

Pengujian dilakukan dengan melibatkan satu responden yang merupakan supervisor selama pelaksanaan kegiatan magang di PT Vastu Cipta Persada. Responden dipilih karena memiliki pemahaman terhadap proses bisnis serta alur penggunaan sistem CAVAA, khususnya pada modul *customer* yang menjadi fokus pengembangan. Sebelum mengisi kuesioner, responden terlebih dahulu mencoba seluruh fitur pada modul *customer*, mulai dari autentikasi, pemindaian *QR Code*, pemilihan menu, pengelolaan keranjang pesanan, *checkout*, hingga melihat detail dan riwayat pesanan. Setelah seluruh proses dicoba, responden diminta mengisi kuesioner SUS yang terdiri dari 10 pernyataan dengan rentang penilaian 1 (Sangat Tidak Setuju) sampai 5 (Sangat Setuju). Hasil jawaban responden dapat dilihat pada Tabel 3.7.

Tabel 3.7 Hasil Pengujian SUS

No	Pertanyaan	Sangat Tidak Setuju			Sangat Setuju	
		1	2	3	4	5
1	Saya merasa sistem pemesanan berbasis <i>QR Code</i> ini layak untuk digunakan kembali pada proses pemesanan makanan dan minuman.					√
2	Menurut saya, alur penggunaan sistem <i>self-order</i> ini masih cukup membingungkan ketika pertama kali digunakan	√				
3	Saya merasa proses pemesanan menu, penambahan produk ke keranjang, dan <i>checkout</i> pada sistem ini mudah dipahami				√	
4	Saya masih membutuhkan bantuan orang lain ketika menggunakan fitur pemesanan pada sistem ini		√			
5	Saya menilai fitur-fitur pada sistem, seperti <i>scan QR Code</i> , keranjang pesanan, <i>checkout</i> , dan status pesanan sudah berjalan dengan baik dan saling terhubung					√
6	Saya merasa masih terdapat beberapa bagian sistem yang kurang konsisten saat digunakan, terutama pada perpindahan halaman dan proses pemesanan		√			

7	Saya yakin pengguna lain dapat mempelajari cara menggunakan sistem <i>self-order</i> ini dengan cepat				√	
8	Menurut saya, tampilan dan alur penggunaan sistem pemesanan ini masih terasa cukup rumit digunakan pada beberapa bagian		√			
9	Saya merasa percaya diri ketika menggunakan sistem ini untuk melakukan pemesanan makanan secara mandiri melalui <i>smartphone</i>				√	
10	Saya perlu mempelajari cara penggunaan sistem terlebih dahulu sebelum dapat menggunakan seluruh fitur pemesanan dengan baik		√			

Berdasarkan jawaban yang diberikan responden pada Tabel 3.7, diperoleh total skor sebesar 33. Nilai *usability* kemudian dihitung menggunakan rumus *System Usability Scale* (SUS), yaitu skor SUS = $(\sum \text{skor}) \times 2,5$ sehingga diperoleh nilai sebesar 82,5. Hasil tersebut menunjukkan bahwa modul *customer* dapat digunakan dengan baik dan fitur-fitur yang tersedia cukup mudah dipahami selama proses pemesanan berlangsung.

BAB IV

REFLEKSI PELAKSANAAN MAGANG

4.1 Relevansi Akademik

Selama pelaksanaan magang di PT Vastu Cipta Persada, penulis terlibat secara langsung dalam pengembangan modul *customer* pada sistem CAVAA yang mendukung layanan *self-order* berbasis *QR Code*. Keterlibatan tersebut memberikan kesempatan untuk menerapkan berbagai materi yang telah dipelajari selama perkuliahan ke dalam pengembangan sistem yang digunakan pada lingkungan kerja. Mata kuliah Pengembangan Aplikasi Berbasis Web menjadi dasar dalam pengembangan aplikasi menggunakan *framework* Laravel, sedangkan mata kuliah Rekayasa Perangkat Lunak dan Pengembangan Sistem Informasi membantu memahami proses analisis kebutuhan, perancangan, implementasi, hingga pengujian sistem. Dengan demikian, kegiatan magang memberikan gambaran mengenai penerapan konsep akademik pada proyek yang dikembangkan untuk memenuhi kebutuhan operasional.

Selama mengikuti perkuliahan, proses pengembangan perangkat lunak dipelajari melalui tahapan yang terstruktur mulai dari analisis kebutuhan, perancangan, implementasi, hingga pengujian. Ketika menjalani kegiatan magang, penulis memahami bahwa pelaksanaan pengembangan di lingkungan kerja tidak selalu berlangsung secara berurutan seperti yang dipelajari di perkuliahan. Perubahan kebutuhan fitur dapat terjadi selama proses pengembangan sehingga rancangan maupun implementasi perlu disesuaikan dengan perkembangan kebutuhan sistem. Selain itu, penerapan *Agile* dengan *Scrum* pada proyek juga tidak sepenuhnya sama dengan teori yang dipelajari. Sebagai contoh, *daily scrum* tidak dilaksanakan setiap hari, tetapi komunikasi mengenai perkembangan pekerjaan tetap dilakukan melalui diskusi bersama supervisor maupun anggota tim sesuai kebutuhan proyek.

Penerapan teori selama kegiatan magang terlihat pada berbagai tugas yang dikerjakan selama pengembangan modul *customer*. Penulis berhasil menyelesaikan pengembangan fitur autentikasi *customer*, implementasi *QR Code* dan manajemen sesi meja, halaman *customer*, keranjang pesanan dan *checkout*, serta halaman detail dan riwayat pesanan sesuai dengan kebutuhan sistem. Selama proses pengembangan, fitur manajemen sesi meja masih memerlukan beberapa penyesuaian berdasarkan hasil diskusi bersama supervisor dan anggota tim agar mekanisme pengelolaan status meja sesuai dengan kebutuhan sistem. Pengalaman tersebut memberikan pemahaman bahwa pengembangan perangkat lunak tidak hanya

bergantung pada penerapan teori, tetapi juga memerlukan proses evaluasi dan penyempurnaan secara berkelanjutan.

Kegiatan magang juga memberikan kesempatan untuk memahami bagaimana berbagai komponen dalam sistem saling terhubung. Pengembangan modul *customer* tidak hanya berfokus pada pembuatan antarmuka, tetapi juga pada keterkaitan antarfitur agar proses pemesanan dapat berjalan dengan baik dari awal hingga akhir. Sebagai contoh, fitur autentikasi *customer*, manajemen sesi meja, *checkout*, serta riwayat pesanan harus saling terintegrasi sehingga proses pemesanan dapat berlangsung dengan lancar. Penerapan *QR Code* juga memberikan pemahaman mengenai pemanfaatan teknologi untuk mendukung kebutuhan operasional, yaitu menghubungkan pelanggan dengan meja yang digunakan sehingga proses pemesanan dapat dilakukan tanpa memasukkan informasi meja secara manual.

Proses pengembangan tersebut juga memberikan pemahaman mengenai pentingnya pengelolaan data dalam sebuah sistem. Data pelanggan, data meja, data produk, data pesanan, dan detail pesanan saling berhubungan untuk mendukung proses bisnis yang berjalan. Dari proses tersebut, penulis memahami bahwa ketepatan pengelolaan data berpengaruh terhadap informasi yang ditampilkan serta kelancaran proses yang berlangsung pada sistem. Penulis juga memperoleh pengalaman dalam proses pengujian setelah pengembangan fitur selesai dilakukan. Setiap fitur diuji menggunakan berbagai skenario untuk memastikan seluruh fungsi berjalan sesuai kebutuhan sistem. Melalui proses tersebut, penulis memahami bahwa pengujian merupakan bagian penting dalam pengembangan perangkat lunak karena membantu menemukan kendala yang masih muncul sebelum sistem digunakan.

Berdasarkan kegiatan magang yang telah dilaksanakan, penulis memperoleh pemahaman yang lebih baik mengenai hubungan antara materi yang dipelajari selama perkuliahan dengan penerapannya di lingkungan kerja. Konsep yang diperoleh di bangku kuliah menjadi dasar dalam menyelesaikan berbagai tugas selama magang, sedangkan pengalaman selama pengembangan sistem memberikan pemahaman bahwa penerapan teori perlu disesuaikan dengan kebutuhan proyek, hasil evaluasi, dan kondisi nyata yang dihadapi selama proses pengembangan.

4.2 Pembelajaran Magang

Selama melaksanakan kegiatan magang di PT Vastu Cipta Persada, penulis memperoleh berbagai pembelajaran yang tidak hanya berkaitan dengan pengembangan perangkat lunak, tetapi juga mengenai cara bekerja dalam lingkungan profesional. Keterlibatan secara langsung

dalam proses pengembangan sistem memberikan pengalaman yang berbeda dibandingkan dengan kegiatan perkuliahan karena setiap pekerjaan harus diselesaikan sesuai kebutuhan proyek serta berkoordinasi dengan supervisor dan anggota tim.

Salah satu pembelajaran yang diperoleh adalah pentingnya memahami kebutuhan sistem sebelum memulai proses pengembangan. Setiap fitur yang dikerjakan diawali dengan pembahasan mengenai kebutuhan pengguna dan tujuan pengembangan sehingga implementasi yang dilakukan dapat sesuai dengan hasil yang diharapkan. Selama proses pengerjaan, penulis juga memahami bahwa kebutuhan sistem dapat berubah berdasarkan hasil evaluasi maupun diskusi bersama supervisor. Pengalaman tersebut mengajarkan penulis bahwa kemampuan untuk menyesuaikan pekerjaan terhadap perubahan kebutuhan merupakan hal yang penting selama kegiatan magang.

Kegiatan magang juga memberikan pengalaman dalam membangun komunikasi yang baik dengan supervisor dan anggota tim. Setiap kendala yang ditemukan selama proses pengembangan didiskusikan terlebih dahulu sebelum dilakukan penyelesaian sehingga solusi yang diterapkan sesuai dengan kebutuhan sistem. Melalui proses tersebut, penulis belajar menyampaikan perkembangan pekerjaan, menerima masukan, serta berdiskusi untuk menentukan penyelesaian yang tepat terhadap permasalahan yang dihadapi.

Selain kemampuan teknis, kegiatan magang juga membantu mengembangkan sikap profesional dalam menyelesaikan pekerjaan. Setiap tugas yang diberikan dikerjakan dengan penuh tanggung jawab serta diselesaikan sesuai target yang telah ditentukan. Penulis juga belajar mengatur waktu agar setiap pekerjaan dapat diselesaikan sesuai dengan jadwal pengembangan yang telah direncanakan. Pengalaman tersebut memberikan pemahaman bahwa kedisiplinan dan tanggung jawab memiliki peran penting dalam mendukung kelancaran pelaksanaan proyek.

Kerja sama antaranggota tim juga menjadi pembelajaran yang berharga selama pelaksanaan magang. Meskipun setiap anggota tim memiliki tanggung jawab masing-masing, proses pengembangan tetap dilakukan melalui koordinasi dan saling berbagi informasi mengenai perkembangan pekerjaan. Kondisi tersebut membantu penulis memahami bahwa keberhasilan suatu proyek tidak hanya ditentukan oleh kemampuan individu, tetapi juga oleh komunikasi, koordinasi, dan kerja sama yang baik antaranggota tim.

Berdasarkan pengalaman selama kegiatan magang, penulis memperoleh gambaran mengenai budaya kerja di lingkungan profesional serta memahami pentingnya kemampuan teknis dan nonteknis dalam mendukung penyelesaian suatu proyek. Pengalaman tersebut

menjadi bekal yang bermanfaat untuk mempersiapkan diri menghadapi dunia kerja setelah menyelesaikan pendidikan di perguruan tinggi.

BAB V

PENUTUP

5.1 Kesimpulan

Kegiatan magang yang dilaksanakan di PT Vastu Cipta Persada telah memberikan kesempatan kepada penulis untuk terlibat secara langsung dalam pengembangan modul *customer* pada sistem *Point of Sale* (POS) CAVAA. Selama kegiatan magang, penulis berhasil mengembangkan fitur *self-order* berbasis *QR Code* dengan manajemen sesi meja yang mendukung proses pemesanan makanan dan minuman secara mandiri.

Fitur yang dikembangkan meliputi autentikasi *customer*, implementasi *QR Code* dan manajemen sesi meja, halaman menu *customer*, keranjang pesanan, *checkout*, serta halaman detail dan riwayat pesanan. Seluruh fitur tersebut telah melalui proses pengujian dan dapat berjalan sesuai dengan kebutuhan sistem. Hasil pengujian fungsional menunjukkan bahwa setiap fitur dapat digunakan dengan baik, sedangkan hasil pengujian *usability* menggunakan metode *System Usability Scale* (SUS) memperoleh skor sebesar 82,5 yang menunjukkan bahwa sistem memiliki tingkat kemudahan penggunaan yang baik.

Selain memperoleh pengalaman dalam pengembangan perangkat lunak, kegiatan magang juga memberikan pemahaman yang lebih mendalam mengenai proses pengembangan sistem pada lingkungan kerja profesional. Melalui kegiatan ini, penulis dapat menerapkan pengetahuan yang dimiliki dalam proses analisis kebutuhan, implementasi fitur, pengelolaan data, serta pengujian sistem sehingga tujuan pelaksanaan magang dan penyusunan laporan ini dapat tercapai dengan baik.

5.2 Saran

Berdasarkan hasil pengembangan dan pengujian yang telah dilakukan, sistem yang dikembangkan telah mampu mendukung proses pemesanan secara mandiri pada modul *customer*. Meskipun demikian, masih terdapat beberapa pengembangan yang dapat dilakukan untuk meningkatkan kualitas layanan yang diberikan oleh sistem.

Pengembangan berikutnya dapat diarahkan pada penambahan fitur yang mendukung pengalaman pengguna, seperti notifikasi status pesanan secara *real-time*. Pengujian sistem juga dapat dilakukan dengan melibatkan lebih banyak responden sehingga hasil evaluasi yang diperoleh dapat memberikan gambaran yang lebih luas mengenai pengalaman pengguna saat menggunakan sistem.

Bagi mahasiswa yang akan melaksanakan magang pada bidang pengembangan perangkat lunak, disarankan untuk tidak hanya berfokus pada penulisan kode program, tetapi juga memahami kebutuhan pengguna, alur bisnis, dan proses komunikasi yang terjadi selama pengembangan sistem berlangsung. Pemahaman tersebut dapat membantu dalam menghasilkan solusi yang lebih sesuai dengan kebutuhan pengguna maupun kebutuhan sistem yang dikembangkan.

DAFTAR PUSTAKA

- Anggoro, D., Sakti, D. V. S. Y., & Waluyo, S. (2021). Implementasi Sistem Informasi Berbasis QR Code Guna Mencegah Kerumunan dalam Antrian Wisuda. *Edumatic: Jurnal Pendidikan Informatika*, 5(1), 128–136. <https://doi.org/10.29408/edumatic.v5i1.3383>
- Asher, A. D., & Hidayat, S. (2024). PENGEMBANGAN SISTEM INFORMASI PENERIMAAN PESERTA DIDIK BARU BERBASIS WEBSITE DENGAN METODE WATERFALL. *EDUSAINTEK: Jurnal Pendidikan, Sains Dan Teknologi*, 11(3), 1485–1502. <https://doi.org/10.47668/edusaintek.v11i3.1135>
- Asra, T., Nur Khasanah, S., & Rikardo Nainggolan, E. (2023). Rancang Bangun Sistem Informasi Manajemen Restoran Berbasis Web Pada Warunk Upnormal. *Reputasi: Jurnal Rekayasa Perangkat Lunak*, 4(2), 110–119. <https://doi.org/10.31294/reputasi.v4i2.2428>
- Dzikria, I., & Rizal, A. (2023). Rancang Bangun Sistem Pemesanan Mandiri Restoran Berbasis Progressive Web Apps. *Jurnal Sistim Informasi Dan Teknologi*, 5(1), 135–144. <https://jsisfotek.org/index.php/JSisfotek/article/view/252>
- Herawati, I. M., & Azahra, D. (2024). EVALUASI USABILITY WEBSITE JASUDA.NET MENGGUNAKAN SYSTEM USABILITY SCALE (SUS). *JUPI (Jurnal Ilmiah Penelitian Dan Pembelajaran Informatika)*, 9(2), 994–1000. <https://doi.org/10.29100/jupi.v9i2.4328>
- Juventauricula, P., Hanggara, B. T., & Pramono, D. (2024). Pengembangan Sistem Informasi Point of Sale (POS) berbasis Web menggunakan Pendekatan Metode Waterfall (Studi Kasus: Restoran Altari). *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, 8(1), 97–106. <https://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/13199>
- Kudadiri, Z. H. (2025). Rancang Bangun Aplikasi Manajemen Reservasi Restoran. *JURNAL MEDIA INFOTAMA*, 21(1), 249–257. <https://doi.org/10.37676/jmi.v21i1.7982>
- Nilawati, L., & Widya, S. A. (2023). Penerapan Metode Scrum Pada Perancangan Sistem Informasi Manajemen Arsip Surat Berbasis Web. *Jurnal Teknologi Dan Sistem Informasi Bisnis*, 5(4), 484–491. <https://doi.org/10.47233/jteksis.v5i4.1044>
- Nova, S. H., Widodo, A. P., & Warsito, B. (2022). Analisis Metode Agile pada Pengembangan Sistem Informasi Berbasis Website: Systematic Literature Review. *Techno.Com*, 21(1), 139–148. <https://doi.org/10.33633/tc.v21i1.5659>
- Pudyawardana, W. (2023). Perancangan Sistem Informasi Pemesanan Makanan Dan Minuman Berbasis Web Pada Restoran Lamongan Cahaya. *ALMUISY: Journal of Al Muslim*

- Information System*, 2(1), 21–27.
<https://journal.almuslim.ac.id/index.php/almuisy/article/view/11>
- Rezeki, S., & Alda, M. (2024). Rancang Bangun Sistem Informasi Manajemen Reservasi Restoran dan Penyewaan Ruangan Berbasis Mobile. *Jutisi: Jurnal Ilmiah Teknik Informatika Dan Sistem Informasi*, 13(2), 1304.
<https://doi.org/10.35889/jutisi.v13i2.2185>
- Rusito, R., & Diyah Intan Mawarni Putri. (2024). Implementasi QR Code Untuk Sistem Informasi Pemesanan Menu Pada Restoran Omah Gedhe Pandean Kaliwungu Berbasis Web. *LogicLink*, 91–105. <https://doi.org/10.28918/logiclink.v1i2.9164>
- Saleh, K. R. (2023). IMPLEMENTASI METODE AGILE SERTA PROSES BISNIS DALAM PENGEMBANGAN DAN PERANCANGAN APLIKASI BERGERAK MECHA SEBAGAI PENYEDIA LAYANAN PERBAIKAN KENDARAAN. *EDUSAINTEK: Jurnal Pendidikan, Sains Dan Teknologi*, 11(1), 87–103.
<https://doi.org/10.47668/edusaintek.v11i1.959>
- Sandfreni, S., Ulum, M. B., & Azizah, A. H. (2021). ANALISIS PERANCANGAN SISTEM INFORMASI PUSAT STUDI PADA FAKULTAS ILMU KOMPUTER UNIVERSITAS ESA UNGGUL. *Sebatik*, 25(2), 345–356. <https://doi.org/10.46984/sebatik.v25i2.1587>
- Susila, A. A. N. H., & Sri Arsa, D. M. (2023). Analisis System Usability Scale (SUS) dan Perancangan Sistem Self Service Pemesanan Menu di Restoran Berbasis Web. *Majalah Ilmiah UNIKOM*, 21(1), 3–8. <https://doi.org/10.34010/miu.v21i1.10683>
- Walim, W., Pohan, A. B., & Safrudin, A. (2023). IMPLEMENTASI METODE AGILE DEVELOPMENT DALAM PERANCANGAN SISTEM INFORMASI PEMESANAN MENU PADA RESTORAN. *Profitabilitas*, 2(2), 106–117.
<https://doi.org/10.31294/profitabilitas.v2i2.1661>
- Wijaya, A., & Wahyuningtyas, E. (2024). Rancang Bangun Sistem Informasi Manajemen Restoran Pada King Fried Chicken. *Jurnal Pesona Informatika*, 1(1 : April 2024), 13–25.
<https://journalng.uwks.ac.id/jpi/article/view/238>
- Yoseph Halim, Sandy Kosasi, Tony Wijaya, & Susanti M. Kuway. (2021). Self-Service Technology Berbasis Android Menggunakan RestFul Web Service Pada Bisnis Restoran. *Journal of Applied Computer Science and Technology*, 2(2), 73–82.
<https://doi.org/10.52158/jacost.v2i2.174>

LAMPIRAN

Logbook Magang

No	Tanggal	Aktivitas
1	15 September – 19 September 2025	Melakukan studi dokumentasi proyek dan memahami kebutuhan sistem sebagai tahap awal pelaksanaan magang.
2	22 September – 6 Oktober 2025	Mengembangkan modul lain dalam proyek CAVAA.
3	7 Oktober – 24 Oktober 2025	Melanjutkan pengembangan pada modul lain dalam proyek CAVAA.
4	27 Oktober – 31 Oktober 2025	Melakukan diskusi bersama supervisor mengenai kebutuhan modul <i>customer</i> , mengembangkan fitur registrasi dan <i>login customer</i> .
5	3 November – 7 November 2025	Mengembangkan fitur <i>QR Code</i> dan manajemen sesi meja serta melakukan penyesuaian berdasarkan hasil diskusi bersama supervisor dan anggota tim.
6	10 November – 14 November 2025	Mengembangkan halaman menu, kategori produk, dan detail produk.
7	17 November – 21 November 2025	Mengembangkan fitur keranjang pesanan, <i>checkout</i> , halaman detail pesanan, dan riwayat pesanan.
8	24 November – 25 November 2025	Menyelesaikan pengembangan modul <i>customer</i> serta melakukan pengujian awal terhadap fitur yang telah dikembangkan.
9	26 November – 5 Desember 2025	Melanjutkan pengembangan pada modul lain dalam proyek CAVAA.
10	9 Desember – 19 Desember 2025	Melakukan pengujian fungsional pada seluruh fitur modul <i>customer</i> .
11	22 Desember – 26 Desember 2025	Melakukan pengujian ulang pada fitur modul <i>customer</i> .
12	29 Desember 2025 – 13 Maret 2026	Melanjutkan pengembangan pada proyek lain.