

**RANCANG BANGUN *FRONT-END* WEB
UNTUK PEMESANAN MAKANAN *ONLINE*
DI KANTIN MENGGUNAKAN REACT JS
(STUDI KASUS: KANTIN UNIVERSITAS PASUNDAN SETIABUDI)**

TUGAS AKHIR

Disusun sebagai salah satu syarat untuk kelulusan
Program Strata 1, Program Studi Teknik Informatika,
Universitas Pasundan Bandung

oleh :

Syifa Rizki Mutia

nrp. 21.304.0005



**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PASUNDAN BANDUNG
SEPTEMBER 2025**

LEMBAR PENGESAHAN
LAPORAN TUGAS AKHIR

Telah diujikan dan disahkan tugas akhir dari :

Nama : Syifa Rizki Mutia

Nrp. : 21.304.0005

Dengan Judul :

“Rancang Bangun *Front-End* Web Untuk Pemesanan Makanan *Online* Di Kantin
Menggunakan React JS (Studi Kasus: Kantin Universitas Pasundan Setiabudi)”

Bandung, 25 September 2025

Menyetujui,
Pembimbing Utama



(Dr. Ayi Purbasari, S.T., M.T)

KATA PENGANTAR

Ucapan dan rasa syukur penulis layangkan ke hadirat Ilahi Robbi, yang telah berkenan menguatkan penulis untuk membuat Laporan Tugas Akhir dengan “Rancang Bangun *Front-End* Web Untuk Pemesanan Makanan *Online* Di Kantin Menggunakan React JS (Studi Kasus: Kantin Universitas Pasundan Setiabudi)”. Adapun penulisan laporan ini bertujuan untuk memenuhi salah satu syarat kelulusan Tugas Akhir, di Program Studi Teknik Informatika Universitas Pasundan.

Penulis menyadari laporan ini dapat terwujud berkat bantuan dan dorongan dari berbagai pihak. Maka pada kesempatan ini penulis sampaikan terima kasih yang sebesar-besarnya atas segala bantuan yang penulis terima baik secara moril maupun materil, sehingga penulis dapat menyelesaikan laporan ini kepada:

1. Allah SWT, atas segala Rahmat, nikmat, dan kemudahan yang diberikan selama proses pengerjaan tugas akhir ini.
2. Kedua orang tua tercinta, yang selalu memberikan doa, kasih sayang, serta dukungan moril dan materil.
3. Dr. Ayi Purbasari, S.T, M.T selaku Dosen Pembimbing, yang telah meluangkan waktu, memberikan bimbingan dan arahan selama proses penyusunan tugas akhir ini.
4. Bapak/Ibu dosen penguji dan seluruh dosen Program Studi Teknik Informatika Universitas Pasundan, yang telah memberikan ilmu dan wawasan selama masa perkuliahan
5. Semua pihak yang tidak dapat penulis sebutkan satu persatu, yang telah membantu secara langsung maupun tidak langsung dalam penyusunan tugas akhir ini.

Dalam penulisan laporan akhir ini, penulis menyadari bahwa masih terdapat banyak kesalahan dan kekeliruan. Oleh karena ini, saran dan kritik yang bersifat membangun akan penulis terima dengan senang hati. Penulis sangat berharap laporan akhir ini dapat bermanfaat bagi semua pihak yang memerlukan.

Akhir kata, semoga penulisan laporan ini dapat bermanfaat bagi penulis dan bagi perkembangan ilmu Teknologi dimasa yang akan datang.

Bandung, 25 September 2025

Penulis

LEMBARAN PERNYATAAN KEASLIAN

LAPORAN TUGAS AKHIR

Saya menyatakan dengan sesungguhnya bahwa:

1. Tugas akhir ini adalah benar-benar asli dan belum pernah diajukan untuk mendapatkan gelar akademik, baik di Universitas Pasundan Bandung maupun di Perguruan Tinggi lainnya.
2. Tugas akhir ini merupakan gagasan, rumusan dan penelitian saya sendiri, tanpa bantuan pihak lain kecuali arahan dari tim Dosen Pembimbing.
3. Dalam tugas akhir ini tidak terdapat karya atau pendapat orang lain, kecuali bagian-bagian tertentu dalam penulisan laporan Tugas Akhir yang saya kutip dari hasil karya orang lain telah dituliskan dalam sumbernya secara jelas sesuai dengan norma, kaidah, dan etika penulisan karya ilmiah, serta disebutkan dalam Daftar Pustaka pada tugas akhir ini.
4. Kakas, perangkat lunak, dan alat bantu kerja lainnya yang digunakan dalam penelitian ini sepenuhnya menjadi tanggung jawab saya, bukan tanggung jawab Universitas Pasundan Bandung.

Apabila di kemudian hari ditemukan seluruh atau sebagian laporan tugas akhir ini bukan hasil karya saya sendiri atau adanya plagiasi dalam bagian-bagian tertentu, saya bersedia menerima sanksi akademik, termasuk pencabutan gelar akademik yang saya sandang sesuai dengan norma yang berlaku di Universitas Pasundan, serta perundang-undangan lainnya.

Bandung, 25 September 2025

Yang membuat pernyataan,

Syifa Rizki Mutia

Nrp. 21.304. 0005

ABSTRAK

Kantin di lingkungan kampus merupakan fasilitas penting yang menjadi tujuan utama bagi mahasiswa dan karyawan untuk menikmati makanan dan minuman dalam menunjang kebutuhan konsumsi mereka. Namun, sistem pemesanan makanan di Kantin Universitas Pasundan Setiabudi masih dilakukan secara manual, yang sering menimbulkan masalah seperti antrean panjang, kesalahan pencatatan pesanan, serta kurangnya informasi menu yang jelas. Permasalahan ini menghambat efisiensi layanan dan mengurangi kenyamanan pengguna dalam melakukan pemesanan makanan.

Penelitian ini berfokus pada pembangunan *front-end* website pemesanan makanan online sebagai solusi digital untuk meningkatkan pengalaman pengguna. Pembangunan dilakukan menggunakan React JS, sebuah pustaka JavaScript yang memungkinkan pembuatan antarmuka berbasis komponen secara efisien dan terstruktur. Dalam proses pembangunan, metode Scrum diterapkan dengan membagi pekerjaan menjadi bagian-bagian kecil yang dikerjakan secara bertahap.

Hasil dari penelitian ini adalah antarmuka *front-end* website yang dilengkapi dengan fitur menu digital dan sistem pemesanan yang responsif. Tampilan antarmuka dirancang agar dapat diakses melalui berbagai perangkat dan memudahkan pengguna dalam memilih serta memesan makanan secara *online*. Dengan penerapan sistem ini, proses pemesanan diharapkan menjadi lebih mudah, mengurangi antrean, serta meningkatkan kenyamanan dan efisiensi transaksi di kantin kampus.

Kata kunci: Kantin kampus, pemesanan makanan *online*, Scrum, *Front-end*, React JS.

ABSTRACT

The campus canteen plays an essential role as a place where students and staff can get food and drinks to support their consumption needs. However, the canteen at Pasundan Setiabudi University still uses a manual food ordering system, which often leads to long queues, incorrect orders, and a lack of clear menu information. These issues reduce service efficiency and make the ordering process less convenient for users.

This research focuses on the development of a front-end interface for an online food ordering website as a digital solution to enhance user experience. The development is carried out using React JS, a JavaScript library that enables the efficient and structured creation of component-based interfaces. During the development process, the Scrum method is applied by dividing the work into smaller parts that are completed incrementally.

The result of this research is a website front-end interface equipped with digital menu features and a responsive ordering system. The interface is designed to be accessible through various devices and makes it easy for users to choose and order food online. With the implementation of this system, the ordering process is expected to become easier, reduce queues, and increase the convenience and efficiency of transactions in the campus canteen.

Keywords: Campus canteen, online food ordering, Scrum, Front-end, React JS.

DAFTAR ISI

ABSTRAK	i
ABSTRACT	ii
DAFTAR ISI	iii
DAFTAR GAMBAR	vi
DAFTAR TABEL.....	x
DAFTAR LAMPIRAN	xiv
DAFTAR SIMBOL	xv
DAFTAR ISTILAH	xvi
BAB 1 PENDAHULUAN	1-1
1.1 Latar Belakang	1-1
1.2 Identifikasi Masalah	1-3
1.3 Tujuan Tugas Akhir	1-3
1.4 Lingkup Tugas Akhir.....	1-3
1.5 Metodologi Tugas Akhir.....	1-4
1.6 Sistematika Penulisan.....	1-6
BAB 2 LANDASAN TEORI DAN PENELITIAN TERDAHULU	2-1
2.1 Teori yang Digunakan	2-1
2.1.1 Website	2-1
2.1.2 Responsive Web Design	2-1
2.1.3 <i>Front-end</i>	2-1
2.1.4 Javascript.....	2-2
2.1.5 React JS	2-3
2.1.6 Tailwind CSS.....	2-4
2.1.7 Scrum.....	2-5
2.1.8 Kantin	2-5
2.1.9 Pemesanan.....	2-5
2.2 Penelitian Terdahulu.....	2-6
BAB 3 SKEMA PENELITIAN	3-1
3.1 Alur Pengerjaan Tugas Akhir	3-1
3.2 Perumusan Masalah.....	3-2

3.2.1	Analisis Sebab Akibat.....	3-2
3.2.2	Analisis Solusi Masalah	3-4
3.3	Profil Penelitian Tugas Akhir	3-5
3.3.1	Deskripsi Singkat.....	3-5
3.3.2	Struktur Organisasi.....	3-6
BAB 4	ANALISIS DAN PERANCANGAN	4-1
4.1	Analisis Kebutuhan Perangkat Lunak	4-1
4.1.1	Analisis Sistem Saat Ini.....	4-1
4.1.2	Analisis Kebutuhan Pengguna.....	4-2
4.1.3	Analisis Kebutuhan Fungsional.....	4-3
4.1.4	Analisis Kebutuhan Non Fungsional.....	4-3
4.2	Pemodelan Berbasis Skenario	4-4
4.2.1	Deskripsi Aktor.....	4-4
4.2.2	Diagram <i>Use Case</i>	4-4
4.2.3	Deskripsi Diagram <i>Use Case</i>	4-6
4.2.4	Skenario <i>Use Case</i>	4-6
4.2.5	Diagram Swimlane	4-22
4.3	Perancangan Kebutuhan Perangkat Lunak	4-34
4.3.1	Pemodelan Data.....	4-34
4.3.2	Pemodelan Berbasis Kelas	4-35
4.3.3	Pemodelan Berbasis Perilaku	4-49
4.3.4	Perancangan Arsitektur Aplikasi	4-63
BAB 5	SCRUM PROCESS DAN DELIVERY	5-1
5.1	Scrum Process	5-1
5.1.1	Product Backlog	5-3
5.1.2	Sprint Ke-1	5-4
5.1.3	Sprint Ke-2	5-13
5.1.4	Sprint Ke-3	5-22
5.1.5	Sprint Ke-4	5-33

5.2 Delivery.....	5-41
5.2.1 Pengujian Perangkat Lunak.....	5-41
5.2.2 Deployment	5-49
BAB 6 KESIMPULAN DAN SARAN	6-1
6.1 Kesimpulan	6-1
6.2 Saran.....	6-1
DAFTAR PUSTAKA	xii

DAFTAR GAMBAR

Gambar 1. 1 Langkah-langkah Penyelesaian Tugas Akhir.....	1-4
Gambar 3. 1 Fishbone Sebab Akibat.....	3-3
Gambar 3. 2 Fishbone Solusi Masalah.....	3-4
Gambar 3. 3 Logo Unpas	3-5
Gambar 3. 4 Struktur Organisasi Kantin Unpas.....	3-6
Gambar 4. 1 Diagram <i>Swimlane</i> Sistem Saat Ini	4-2
Gambar 4. 2 Diagram <i>Use Case</i> Pembeli.....	4-4
Gambar 4. 3 Diagram <i>Use Case</i> Pembeli.....	4-5
Gambar 4. 4 Diagram <i>Use Case</i> Penjual.....	4-5
Gambar 4. 5 Wireframe Halaman Beranda	4-7
Gambar 4. 6 Wireframe Halaman Kios.....	4-7
Gambar 4. 7 Wireframe Halaman Kios.....	4-8
Gambar 4. 8 Wireframe Halaman Menu	4-8
Gambar 4. 9 Wireframe Halaman Menu	4-9
Gambar 4. 10 Wireframe Halaman Detail Menu	4-9
Gambar 4. 11 Wireframe Halaman Menu	4-10
Gambar 4. 12 Wireframe Menampilkan Menu Berdasarkan Kata Kunci	4-10
Gambar 4. 13 Wireframe Halaman Menu	4-11
Gambar 4. 14 Wireframe Halaman Keranjang.....	4-12
Gambar 4. 15 Wireframe Halaman Keranjang.....	4-12
Gambar 4. 16 Wireframe Halaman Konfirmasi Detail Pesanan	4-13
Gambar 4. 17 Wireframe Halaman Konfirmasi Detail Pesanan Ketika Tombol Bayar Aktif.....	4-14
Gambar 4. 18 Wireframe Pop Up Pembayaran Berhasil.....	4-14
Gambar 4. 19 Wireframe Pop Up Pembayaran Berhasil.....	4-15
Gambar 4. 20 Wireframe Halaman Status Pesanan.....	4-15
Gambar 4. 21 Wireframe Halaman Status Pesanan Dengan Status Sedang Diproses.....	4-16
Gambar 4. 22 Wireframe Halaman Status Pesanan Dengan Status Selesai Dan Tombol Aktif	4-16
Gambar 4. 23 Wireframe Halaman Dashboard	4-17
Gambar 4. 24 Wireframe Daftar Menu Dashboard	4-17
Gambar 4. 25 Wireframe Halaman Pesanan Dengan Pesanan Kosong.....	4-18
Gambar 4. 26 Wireframe Halaman Pesanan Dengan Daftar Pesanan Masuk	4-18
Gambar 4. 27 Wireframe Halaman Pesanan	4-19
Gambar 4. 28 Wireframe Halaman Detail Pesanan.....	4-19

Gambar 4. 29 Wiraframe Halaman Detail Pesanan Dengan Status Menunggu Diproses	4-20
Gambar 4. 30 Wiraframe Halaman Detail Pesanan Dengan Status Pesanan Diproses	4-20
Gambar 4. 31 Wiraframe Halaman Detail Pesanan Dengan Status Pesanan Diproses	4-21
Gambar 4. 32 Wireframe Halaman Riwayat Pesanan	4-21
Gambar 4. 33 Diagram Swimlane - Menampilkan Daftar Kios.....	4-22
Gambar 4. 34 Diagram Swimlane - Menampilkan Daftar Menu Perkios	4-23
Gambar 4. 35 Diagram <i>Swimlane</i> - Menampilkan Halaman Detail Menu.....	4-23
Gambar 4. 36 Diagram <i>Swimlane</i> - Mencari Menu.....	4-24
Gambar 4. 37 Diagram <i>Swimlane</i> - Mengelola Keranjang	4-25
Gambar 4. 38 Diagram <i>Swimlane</i> - Membuat Pesanan Dari Keranjang	4-26
Gambar 4. 39 Diagram <i>Swimlane</i> - Menyediakan <i>Payment Gateway</i>	4-27
Gambar 4. 40 Diagram <i>Swimlane</i> - Menampilkan Status dan Estimasi Waktu.....	4-28
Gambar 4. 41 Diagram <i>Swimlane</i> - Menerima Notifikasi Pesanan Selesai	4-29
Gambar 4. 42 Diagram <i>Swimlane</i> - Mengelola Menu.....	4-30
Gambar 4. 43 Diagram <i>Swimlane</i> - Menerima Notifikasi Pesanan Masuk.....	4-31
Gambar 4. 44 Diagram <i>Swimlane</i> - Menampilkan Daftar Pesanan dan Detail Pesanan	4-31
Gambar 4. 45 Diagram <i>Swimlane</i> - Mengubah Status Pesanan	4-32
Gambar 4. 46 Diagram <i>Swimlane</i> - Menampilkan Tipe Pengantaran	4-33
Gambar 4. 47 Diagram Perancangan Kelas.....	4-49
Gambar 4. 48 Diagram Sequence - Menampilkan Daftar Kios.....	4-50
Gambar 4. 49 Diagram Sequence - Menampilkan Daftar Menu Perkios.....	4-51
Gambar 4. 50 Diagram Sequence - Menampilkan Detail Menu	4-51
Gambar 4. 51 Diagram Sequence - Mencari Menu.....	4-52
Gambar 4. 52 Diagram Sequence - Mengelola Keranjang.....	4-53
Gambar 4. 53 Diagram Sequence - Membuat Pesanan dari Keranjang	4-54
Gambar 4. 54 Diagram Sequence - Menyediakan <i>Payment Gateway</i>	4-55
Gambar 4. 55 Diagram Sequence - Menampilkan status dan estimasi waktu pesanan.....	4-56
Gambar 4. 56 Diagram Sequence - Menerima Notifikasi Pesanan Selesai.....	4-57
Gambar 4. 57 Diagram Sequence - Mengelola Menu	4-58
Gambar 4. 58 Diagram Sequence - Menerima Notifikasi Pesanan Masuk	4-59
Gambar 4. 59 Diagram Sequence - Menampilkan Daftar Pesanan dan Detail Pesanan	4-60
Gambar 4. 60 Diagram Sequence - Mengubah Status Pesanan.....	4-61
Gambar 4. 61 Diagram Sequence - Menampilkan Tipe Pengantaran	4-62
Gambar 4. 62 Perancangan Arsitektur Aplikasi	4-63

Gambar 5. 1 Struktur Folder	5-2
Gambar 5. 2 Konfigurasi Penjual.....	5-3
Gambar 5. 3 Konfigurasi Pembeli.....	5-3
Gambar 5. 4 Atarmuka Halaman Registrasi Penjual.....	5-6
Gambar 5. 5 Atarmuka Halaman Registrasi Kios	5-6
Gambar 5. 6 Atarmuka Halaman Verifikasi OTP.....	5-6
Gambar 5. 7 Atarmuka Halaman Login	5-7
Gambar 5. 8 Atarmuka Halaman Lupa Kata Sandi.....	5-7
Gambar 5. 9 Atarmuka Halaman Reset Kata Sandi	5-7
Gambar 5. 10 Atarmuka Edit Profil Penjual.....	5-8
Gambar 5. 11 Atarmuka Edit Profil Toko.....	5-8
Gambar 5. 12 Atarmuka Halaman Daftar Menu	5-9
Gambar 5. 13 Atarmuka Halaman Tambah Menu.....	5-9
Gambar 5. 14 Atarmuka Halaman Edit Menu.....	5-9
Gambar 5. 15 Atarmuka Popup Hapus Menu.....	5-10
Gambar 5. 16 Atarmuka Detail Menu	5-10
Gambar 5. 17 Atarmuka Beranda - Hero dan Card Slider.....	5-11
Gambar 5. 18 Atarmuka Beranda - Kios Card	5-11
Gambar 5. 19 Atarmuka Halaman Kios	5-11
Gambar 5. 20 Atarmuka Halaman Menu Per Kios.....	5-12
Gambar 5. 21 Atarmuka Search Bar.....	5-12
Gambar 5. 22 Integrasi Halaman Registrasi Penjual.....	5-15
Gambar 5. 23 Integrasi Halaman Registrasi Kios	5-15
Gambar 5. 24 Integrasi Halaman Verifikasi OTP.....	5-16
Gambar 5. 25 Integrasi Halaman Login	5-16
Gambar 5. 26 Integrasi Halaman Lupa Kata Sandi.....	5-16
Gambar 5. 27 Integrasi Halaman Reset Kata Sandi	5-17
Gambar 5. 28 Integrasi Edit Profil Penjual	5-17
Gambar 5. 29 Integrasi Halaman Edit Profil Toko.....	5-17
Gambar 5. 30 Integrasi Halaman Daftar Menu	5-18
Gambar 5. 31 Integrasi Berhasil Menambahkan Menu Baru	5-18
Gambar 5. 32 Integrasi Berhasil Mengubah Menu	5-18
Gambar 5. 33 Integrasi Menghapus Menu	5-19
Gambar 5. 34 Integrasi Detail Menu	5-19
Gambar 5. 35 Integrasi Halaman Kios	5-20

Gambar 5. 36 Integrasi Halaman Menu Per Kios	5-20
Gambar 5. 37 Atarmuka dan Integrasi Halaman Detail Menu	5-21
Gambar 5. 38 Atarmuka dan Integrasi Keranjang.....	5-21
Gambar 5. 39 Integrasi Beranda - Card Slider Menu Terbaru.....	5-24
Gambar 5. 40 Integrasi Beranda - Kios Terbaru.....	5-25
Gambar 5. 41 Integrasi Beranda - Mencari Menu dan Kios.....	5-25
Gambar 5. 42 Integrasi Keranjang - Menampilkan Item Keranjang	5-26
Gambar 5. 43 Integrasi Keranjang - Mengubah Jumlah Item Keranjang.....	5-26
Gambar 5. 44 Integrasi Keranjang - Menghapus Item Keranjang.....	5-26
Gambar 5. 45 Antarmuka dan Integrasi Halaman Konfirmasi Detail Pesanan	5-27
Gambar 5. 46 Integrasi Menampilkan Tipe Pengantaran	5-27
Gambar 5. 47 Antarmuka <i>Payment Gateway</i>	5-28
Gambar 5. 48 Antarmuka dan Integrasi Halaman Daftar Pesanan.....	5-29
Gambar 5. 49 Antarmuka dan Integrasi Halaman Detail Pesanan	5-29
Gambar 5. 50 Antarmuka Halaman Status Pesanan	5-29
Gambar 5. 51 Antarmukan Status Menunggu Diproses	5-30
Gambar 5. 52 Antarmuka Status Pesanan Diproses	5-30
Gambar 5. 53 Antarmuka Status Pesanan Diantar	5-31
Gambar 5. 54 Antarmuka Pesanan Selesai.....	5-31
Gambar 5. 55 Integrasi Pesanan Masuk	5-32
Gambar 5. 56 Integrasi Halaman Konfirmasi Detail Pesanan.....	5-35
Gambar 5. 57 Integrasi <i>Payment Gateway</i>	5-35
Gambar 5. 58 Antarmuka dan Integrasi Halaman Daftar Peasanan - Pagination.....	5-36
Gambar 5. 59 Antarmuka dan Integrasi Halaman Status Pesanan - Status Menunggu Diproses.....	5-36
Gambar 5. 60 Antarmuka Halaman Status Pesanan - Alert Pemberitahuan.....	5-37
Gambar 5. 61 Antarmuka dan Integrasi Halaman Status Pesanan - Status Sedang Diproses	5-37
Gambar 5. 62 Antarmuka Halaman Status Pesanan - Alert Pesanan Terlambat.....	5-37
Gambar 5. 63 Integrasi Satatus Pesanan - Menunggu Diproses.....	5-38
Gambar 5. 64 Integrasi Satatus Pesanan - Pesanan Diproses	5-38
Gambar 5. 65 Integrasi Satatus Pesanan - Konfirmasi Tipe Pengantaran	5-39
Gambar 5. 66 Integrasi Satatus Pesanan - Siap Diambil.....	5-39
Gambar 5. 67 Integrasi Satatus Pesanan - Pesanan selesai	5-39
Gambar 5. 68 Antarmuka Pesanan Selesai.....	5-40
Gambar 5. 69 Hasil Deployment Dashboard Penjual.....	5-50
Gambar 5. 70 Hasil Deployment Aplikasi Pembeli	5-51

DAFTAR TABEL

Tabel 2. 1 Penelitian Terdahulu	2-6
Tabel 3. 1 Alur Pengerjaan Tugas Akhir	3-1
Tabel 3. 2 Detail Fishbone Sebab Akibat	3-3
Tabel 3. 3 Detail Fishbone Solusi Masalah	3-4
Tabel 4. 1 Diagram <i>Swimlane</i> Analisis Sistem Saat Ini.....	4-2
Tabel 4. 2 <i>User Requirement</i>	4-3
Tabel 4. 3 <i>Software Requirement</i>	4-3
Tabel 4. 4 Analisis Kebutuhan Non Fungsional.....	4-3
Tabel 4. 5 Deskripsi Aktor.....	4-4
Tabel 4. 6 Diagram <i>Use Case</i> Pembeli.....	4-4
Tabel 4. 7 Diagram <i>Use Case</i> Penjual	4-5
Tabel 4. 8 Deskripsi Diagram <i>Use Case</i> Pembeli.....	4-6
Tabel 4. 9 Deskripsi Diagram <i>Use Case</i> Penjual.....	4-6
Tabel 4. 10 Skenario UC-01	4-6
Tabel 4. 11 Skenario UC-02	4-7
Tabel 4. 12 Skenario UC-03	4-9
Tabel 4. 13 Skenario UC-04.....	4-10
Tabel 4. 14 Skenario UC-05	4-11
Tabel 4. 15 Skenario UC-06.....	4-12
Tabel 4. 16 Skenario UC-07	4-13
Tabel 4. 17 Skenario UC-08.....	4-14
Tabel 4. 18 Skenario UC-09.....	4-15
Tabel 4. 19 Skenario UC-10.....	4-16
Tabel 4. 20 Skenario UC-11	4-17
Tabel 4. 21 Skenario UC-12	4-18
Tabel 4. 22 Skenario UC-13	4-19
Tabel 4. 23 Skenario UC-14.....	4-20
Tabel 4. 24 Diagram <i>Swimlane</i> UC-01	4-22
Tabel 4. 25 Diagram <i>Swimlane</i> UC-02	4-23
Tabel 4. 26 Diagram <i>Swimlane</i> UC-03	4-23
Tabel 4. 27 Diagram <i>Swimlane</i> UC-04	4-24
Tabel 4. 28 Diagram <i>Swimlane</i> UC-05	4-25
Tabel 4. 29 Diagram <i>Swimlane</i> UC-06	4-26

Tabel 4. 30 Diagram Swimlane UC-07	4-27
Tabel 4. 31 Diagram Swimlane UC-08	4-28
Tabel 4. 32 Diagram Swimlane UC-09	4-29
Tabel 4. 33 Diagram Swimlane UC-10	4-30
Tabel 4. 34 Diagram Swimlane UC-11	4-31
Tabel 4. 35 Diagram Swimlane UC-12	4-31
Tabel 4. 36 Diagram Swimlane UC-13	4-32
Tabel 4. 37 Diagram Swimlane UC-14	4-33
Tabel 4. 38 Objek Data.....	4-34
Tabel 4. 39 Atribut Data.....	4-34
Tabel 4. 40 Identifikasi Kelas Analisis	4-36
Tabel 4. 41 Identifikasi Atribut.....	4-37
Tabel 4. 42 Identifikasi Operasi	4-43
Tabel 4. 43 Diagram Perancangan Kelas.....	4-49
Tabel 4. 44 Pemodelan Berbasis Perilaku	4-49
Tabel 4. 45 Diagram Sequence UC-01	4-50
Tabel 4. 46 Diagram Sequence UC-02	4-51
Tabel 4. 47 Diagram Sequence UC-03	4-51
Tabel 4. 48 Diagram Sequence UC-04	4-52
Tabel 4. 49 Diagram Sequence UC-05	4-53
Tabel 4. 50 Diagram Sequence UC-06	4-54
Tabel 4. 51 Diagram Sequence UC-07	4-55
Tabel 4. 52 Diagram Sequence UC-08	4-56
Tabel 4. 53 Diagram Sequence UC-09	4-57
Tabel 4. 54 Diagram Sequence UC-10	4-58
Tabel 4. 55 Diagram Sequence UC-11	4-59
Tabel 4. 56 Diagram Sequence UC-12	4-60
Tabel 4. 57 Diagram Sequence UC-13	4-61
Tabel 4. 58 Diagram Sequence UC-14	4-62
Tabel 5. 1 Kakas Perangkat Lunak.....	5-1
Tabel 5. 2 Struktur Folder	5-2
Tabel 5. 3 Konfigurasi.....	5-3
Tabel 5. 4 Product Backlog	5-3
Tabel 5. 5 Sprint Planning Ke-1	5-4
Tabel 5. 6 Sprint Backlog Ke-1	5-4

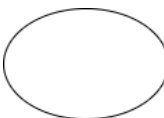
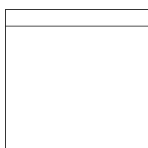
Tabel 5. 7 Increment - Autentikasi (UI)	5-6
Tabel 5. 8 Increment - Mengelola Menu (UI)	5-9
Tabel 5. 9 Increment - Beranda (UI)	5-11
Tabel 5. 10 Increment - Menampilkan daftar kios (UI).....	5-11
Tabel 5. 11 Increment - Menampilkan daftar menu perkios (UI).....	5-12
Tabel 5. 12 Increment - Mencari menu (UI)	5-12
Tabel 5. 13 Sprint Review Ke-1	5-13
Tabel 5. 14 Sprint Planning Ke-2	5-14
Tabel 5. 15 Sprint Backlog Ke-2	5-14
Tabel 5. 16 Increment - Autentikasi (Integrasi).....	5-15
Tabel 5. 17 Increment - Mengelola Menu (Integrasi).....	5-18
Tabel 5. 18 Increment - Menampilkan daftar kios (Integrasi).....	5-20
Tabel 5. 19 Increment - Menampilkan daftar menu per kios (Integrasi).....	5-20
Tabel 5. 20 Increment - Menampilkan detail menu (UI dan Integrasi).....	5-21
Tabel 5. 21 Increment - Mengelola keranjang (UI dan Integrasi)	5-21
Tabel 5. 22 Sprint Review Ke-2	5-22
Tabel 5. 23 Sprint Planning Ke-3	5-23
Tabel 5. 24 Sprint Backlog Ke-3	5-23
Tabel 5. 25 Increment - Beranda (Integrasi).....	5-24
Tabel 5. 26 Increment - Mengelola keranjang (Integrasi).....	5-26
Tabel 5. 27 Increment - Membuat pesanan dari keranjang (UI dan Integrasi).....	5-27
Tabel 5. 28 Increment - Menampilkan tipe pengantaran (Integrasi)	5-27
Tabel 5. 29 Increment - Menyediakan Payment gateway.....	5-28
Tabel 5. 30 Increment - Menampilkan daftar pesanan dan detail pesanan (UI dan Integrasi)	5-29
Tabel 5. 31 Increment - Menampilkan status dan estimasi waktu pesanan (UI dan Integrasi)	5-29
Tabel 5. 32 Increment - Mengubah status pesanan (UI dan Integrasi).....	5-30
Tabel 5. 33 Increment - Mengirim notifikasi pesanan masuk (Integrasi).....	5-32
Tabel 5. 34 Sprint Review Ke-3	5-32
Tabel 5. 35 Sprint Planning Ke-4	5-33
Tabel 5. 36 Sprint Backlog Ke-4	5-34
Tabel 5. 37 Increment - Membuat pesanan dari keranjang (Perbaikan).....	5-35
Tabel 5. 38 Increment - Menyediakan Payment gateway (Perbaikan)	5-35
Tabel 5. 39 Increment - Menampilkan daftar pesanan dan detail pesanan (Perbaikan)	5-36
Tabel 5. 40 Increment - Menampilkan status dan estimasi waktu pesanan (UI dan Integrasi)	5-36
Tabel 5. 41 Increment - Mengubah status pesanan (Integrasi).....	5-38

Tabel 5. 42 Increment - Mengirim notifikasi pesanan selesai (UI)	5-40
Tabel 5. 43 Sprint Review Ke-4	5-40
Tabel 5. 44 Pengujian Autentikasi	5-41
Tabel 5. 45 Pengujian Menampilkan Kios dan Menu	5-44
Tabel 5. 46 Pengujian Mengelola Keranjang	5-44
Tabel 5. 47 Pengujian Membuat Pesanan.....	5-45
Tabel 5. 48 Pengujian <i>Payment Gateway</i>	5-46
Tabel 5. 49 Pengujian Mengubah status Pesanan.....	5-46
Tabel 5. 50 Pengujian Mengelola Menu.....	5-46
Tabel 5. 51 Pengujian Mengirim Notifikasi Pesanan Masuk	5-47
Tabel 5. 52 Pengujian Menampilkan Daftar dan Detail Pesanan	5-47
Tabel 5. 53 Pengujian Menampilkan Tipe Pengantaran	5-48
Tabel 5. 54 Pengujian Halaman Beranda	5-48
Tabel 5. 55 Rekapitulasi Pengujian	5-49
Tabel 5. 56 Hasil Deployment.....	5-50

DAFTAR LAMPIRAN

Gambar A. 1 Berita Acara Wawancara Dengan Pembeli (1).....	A-1
Gambar A. 2 Berita Acara Wawancara Dengan Pembeli (2).....	A-2
Gambar A. 3 Berita Acara Wawancara Dengan Pembeli (3).....	A-3
Gambar A. 4 Berita Acara Wawancara Dengan Penjual (1).....	A-4
Gambar A. 5 Berita Acara Wawancara Dengan Penjual (2).....	A-5
Gambar A. 6 Berita Acara Wawancara Dengan Penjual (3).....	A-6
Gambar A. 7 Berita Acara Wawancara Dengan Badan Pengelola Kantin.....	A-7
Gambar B. 1 Dokumentasi Dengan Badan Pengelolala Kantin.....	B-1
Gambar B. 2 Dokumentasi Dengan Penjual.....	B-1
Gambar B. 3 Dokumentasi Dengan Pembeli.....	B-2
Gambar C. 1 Dokumentasi Sprint 1	C-1
Gambar C. 2 Dokumentasi Sprint 2	C-1
Gambar C. 3 Dokumentasi Sprint 3	C-2
Gambar C. 4 Dokumentasi Sprint 4	C-2
Gambar D. 1 Berita Acara Delivery Penjual	D-1
Gambar D. 2 Berita Acara Delivery Pembeli	D-2
Gambar D. 3 Dokumentasi Delivery Penjual.....	D-3
Gambar D. 4 Dokumentasi Delivery Pembeli.....	D-4

DAFTAR SIMBOL

No	Simbol	Nama	Deskripsi
1.		Start State	Merepresentasikan titik awal dimulainya suatu aktivitas dalam diagram aktivitas.
2.		End State	Merepresentasikan titik akhir dari suatu aktivitas dalam diagram aktivitas.
3.		Activity	Menggambarkan aktifitas yang sedang dilakukan.
4.		Control Flow	Menghubungkan aktivitas dan menunjukkan urutan alur proses.
5.		Actor	Menjelaskan peran yang dijalankan oleh pengguna saat berinteraksi use case.
6.		Use Case	Menjelaskan aktifitas yang dilakukan oleh aktor
7.		Association	Menghubungkan antara aktor dan use case, untuk menggambarkan hubungan/interaksi.
8.		Swimlane	Pembagian dalam diagram aktivitas untuk mengelompokkan aktivitas berdasarkan peran

DAFTAR ISTILAH

No	Istilah	Deskripsi
1.	<i>Scrum</i>	Metodologi manajemen proyek Agile, digunakan untuk mengembangkan produk secara bertahap dan berulang untuk memungkinkan penyempurnaan terus-menerus. Dalam Scrum, pekerjaan dibagi menjadi sprint dan melibatkan peran seperti Scrum Master, Product Owner, dan tim pengembang.
2.	<i>Front-End</i>	Bagian pembangunan aplikasi atau website yang fokus pada pembuatan tampilan visual dan interaksi pengguna, biasanya menggunakan HTML, CSS, dan JavaScript.
3.	<i>React JS</i>	Library JavaScript untuk membangun antarmuka pengguna (UI) berbasis komponen yang dinamis dan interaktif, dikembangkan oleh Facebook.
4.	<i>Tailwind CSS</i>	Framework CSS yang menggunakan pendekatan utility-first, memungkinkan developer membuat desain cepat dengan menambahkan kelas-kelas kecil di dalam HTML tanpa menulis CSS manual.
5.	<i>Fishbone</i>	Diagram yang digunakan untuk menemukan sumber utama masalah dengan mengelompokkan penyebab ke dalam kategori.

BAB 1

PENDAHULUAN

Bab ini menjelaskan tentang pandangan awal persoalan yang terjadi dalam penulisan laporan tugas akhir, berisi latar belakang, identifikasi masalah, tujuan tugas akhir, lingkup tugas akhir, metodologi tugas akhir dan sistematika tugas akhir.

1.1 Latar Belakang

Perkembangan teknologi yang sangat pesat saat ini memberikan kemudahan bagi manusia dalam berbagai aspek kehidupan. Semua kalangan kini dapat dengan mudah mengakses dan memanfaatkan teknologi yang tersedia. Kehadiran teknologi telah menjadi bagian penting dalam aktivitas sehari-hari, bahkan telah mengubah cara kita berkomunikasi, bekerja, belajar, dan bersosialisasi. Di era *modern* ini, akses terhadap informasi menjadi semakin mudah berkat *internet*, salah satunya melalui penggunaan website [NEY24].

Website merupakan sebuah sistem informasi yang memberikan beragam layanan dan informasi melalui internet, yang berperan sebagai sarana efektif dalam membantu pengguna mendapatkan informasi yang mereka butuhkan. Dengan adanya kemudahan akses terhadap berbagai fitur yang mendukung kebutuhan pengguna, seperti pencarian produk, jasa, atau layanan tertentu, website menjadi lebih efisien dan dapat diakses dengan mudah melalui berbagai perangkat yang terhubung ke internet [TJA24].

Salah satu bagian penting dari sebuah website adalah *front-end*, yaitu tampilan antarmuka yang memungkinkan terjadinya interaksi langsung antara pengguna dengan website. *Front-end* menyajikan semua informasi dan fitur yang ada pada sistem dalam bentuk visual yang mudah dimengerti. *Front-end* juga sangat penting dalam menciptakan pengalaman pengguna yang nyaman dan menarik saat berinteraksi dengan website. Seseorang yang membuat tampilan sebuah website disebut dengan *Front-end Developer*, yang bertanggung jawab untuk membuat tampilan sebuah website dan aplikasi menjadi lebih interaktif, menarik dan ramah pengguna [CHA20].

Untuk mendukung pembangunan *front-end*, digunakan pustaka javascript yaitu React JS yang dikembangkan oleh facebook. React JS merupakan pustaka javascript yang bersifat *open-source* yang digunakan untuk membangun antarmuka pengguna pada aplikasi berbasis website. Karena bersifat *open-source*, banyak *library* yang dapat digunakan di internet untuk mendukung proses pengembangan aplikasi, dan react js menggunakan pendekatan berbasis komponen yang memungkinkan pengembang untuk membagi antarmuka pengguna menjadi bagian-bagian kecil yang dapat digunakan kembali. Dengan React JS, pengembang dapat membangun aplikasi web yang responsive, interaktif, dan mudah dalam proses pemeliharaan kode [MUS24]. Kantin Universitas Pasundan (Unpas) di Kampus Setiabudi adalah tempat yang menjadi tujuan utama bagi mahasiswa dan karyawan untuk menikmati makanan dan minuman.

Aktivitas jual beli berlangsung padat, terutama pada jam-jam istirahat. Namun, proses pemesanan makanan di kantin Unpas masih dilakukan secara manual, sehingga penjual harus melayani pembeli satu per satu, yang sering menyebabkan antrean panjang dan waktu tunggu yang lama. Terkadang pesanan yang dibuat juga tidak sesuai dengan keinginan karena kesalahan dalam pencatatan, dan pembeli tidak mengetahui estimasi waktu penyelesaian pesanan mereka, yang dapat menimbulkan ketidakpastian dan rasa tidak nyaman saat menunggu. Selain itu, penjual sering mengalami kebingungan dalam mengelola pesanan pelanggan dan mengatur data menu yang dapat berubah-ubah. Kondisi ini dapat menurunkan kepuasan pelanggan dan menghambat operasional kantin secara keseluruhan.

Penelitian yang dilakukan oleh Dea Syah Kania, Dina Marwah Afirahmi, Dwindi Putri Nuria, dan Dadang Yusup pada tahun 2023 membahas mengenai pembuatan aplikasi E(AT)VERYDAY sebagai Sistem Pemesanan Kantin Berbasis Website untuk meningkatkan pengalaman dalam proses transaksi di kantin. Penelitian ini bertujuan untuk melakukan transaksi memesan makanan secara online dengan efisiensi dan kenyamanan yang lebih tinggi. Hasil penelitian menunjukkan bahwa aplikasi yang dihasilkan dapat memberikan kemudahan bagi admin dan pembeli untuk bertransaksi, admin memiliki kemudahan dalam memasukkan, mengedit, dan menghapus informasi yang diperlukan, seperti melakukan pembaruan pesanan dengan cepat, Pembeli memiliki kebebasan untuk mencari menu yang diinginkan melalui fitur *search bar* sebelum memesan [KAN23].

Penelitian yang dilakukan oleh Luki Rizki dan Yuyun Tresnawati pada tahun 2024 membahas mengenai pembuatan aplikasi pemesanan makanan dan minuman di Cafe Senja untuk meningkatkan efisiensi pelayanan dan pengalaman saat belanja. Penelitian ini bertujuan sebagai alat bantu bagi pelanggan agar lebih mudah dalam melakukan pemesanan dan pelayan serta pemilik cafe dalam pengelolaan pesanan. Hasil penelitian ini adalah aplikasi mampu memudahkan pelanggan pada saat melakukan pemesanan serta memberikan informasi terkait menu, mampu memudahkan pelayan dan pemilik cafe dalam mengelola pesanan [RIZ24].

Berdasarkan permasalahan yang ada di kantin Unpas, penelitian ini mengusulkan, pembangunan website pemesanan makanan online. Dengan adanya sistem pemesanan *online*, pelanggan dapat memesan makanan tanpa harus mengantri. Website ini juga membuat menu lebih mudah dilihat dan memudahkan pelanggan memilih sesuai keinginan, sehingga mengurangi kesalahan pesanan dan juga dapat melakukan pembayaran secara langsung melalui aplikasi. Selain itu, penjual dapat mengelola pesanan dengan lebih terorganisir, memastikan setiap pesanan tercatat dan dikirimkan dengan tepat. Pada penelitian ini berfokus pada pembangunan *front-end* menggunakan bahasa pemrograman JavaScript dengan React Js sebagai pustaka yang digunakan.

1.2 Identifikasi Masalah

Berdasarkan latar belakang diatas, maka dapat diperoleh identifikasi masalah sebagai berikut :

1. Bagaimana membangun antarmuka website pemesanan makanan yang intuitif dan mudah digunakan agar pelanggan dapat melakukan pemesanan dengan lebih cepat dan mudah?
2. Bagaimana mengimplementasikan *library* React JS untuk membangun *front-end* website pemesanan makanan yang responsif, mudah digunakan, dan efisien dalam pengelolaannya?

1.3 Tujuan Tugas Akhir

Berdasarkan identifikasi masalah tersebut di atas, maka dapat diperoleh tujuan tugas akhir yaitu:

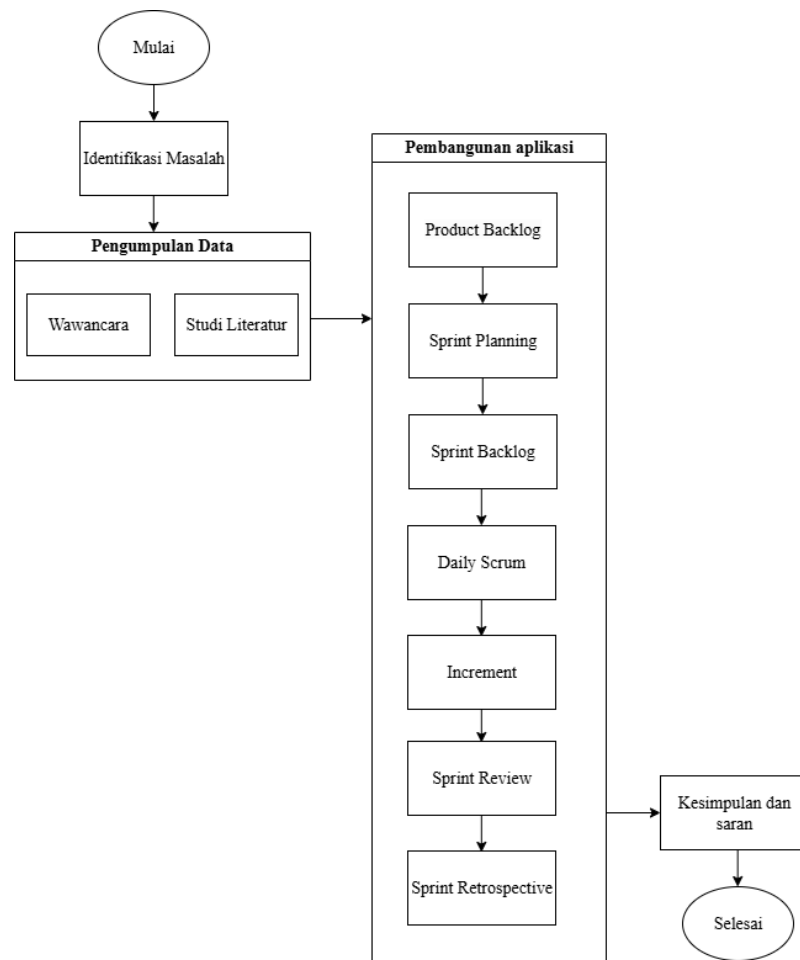
1. Membangun antarmuka website pemesanan makanan yang intuitif dan mudah digunakan, sehingga pelanggan dapat melakukan pemesanan dengan cepat dan mudah.
2. Mengimplementasikan *library* React JS untuk membangun *front-end* website pemesanan makanan yang responsif, mudah digunakan, dan efisien dalam pengelolaannya, sehingga kode yang dihasilkan lebih terstruktur, mudah dipahami, dan dapat dikembangkan lebih lanjut.

1.4 Lingkup Tugas Akhir

Berikut dibawah ini merupakan lingkup dari tugas akhir :

1. Fokus penelitian ini adalah pada pembangunan *Front-End*.
2. Pembangunan aplikasi ini menggunakan *library* React JS dan framework CSS Tailwind CSS.

1.5 Metodologi Tugas Akhir



Gambar 1. 1 Langkah-langkah Penyelesaian Tugas Akhir

Berikut ini merupakan penjelasan dari metodologi pengerjaan tugas akhir :

1. Identifikasi Masalah

Pada tahap identifikasi masalah dilakukan untuk mengetahui masalah apa saja yang ada di kantin Universitas Pasundan Setiabudi.

2. Pengumpulan Data

Pada tahap pengumpulan data ini dilakukan untuk mengetahui pengalaman pengguna dalam menggunakan layanan kantin Universitas Pasundan secara manual. Adapun cara-cara pengumpulan data yang dilakukan adalah sebagai berikut:

a. Wawancara

Tahap wawancara ini dilakukan untuk mendapatkan informasi dari pihak yang terlibat dalam proses pemesanan makanan di kantin Universitas Pasundan Setiabudi, seperti staf kantin dan mahasiswa melalui tanya jawab secara langsung.

b. Studi Literatur

Studi literatur dilakukan untuk mencari teori terkait Pembangunan *Front-End* untuk pemesanan makanan online dengan menggunakan metodologi Scrum dan juga penelitian terdahulu sebagai referensi dalam pengerjaan tugas akhir, yang dapat bersumber dari jurnal ilmiah, artikel-artikel ilmiah dan buku yang berkaitan dengan penelitian yang dilakukan.

3. Metode Scrum

Penulis menggunakan Metode Scrum dalam Pembangunan website pemesanan makanan online ini, terdiri dari beberapa tahap berikut [SCH20]:

a. Product Backlog

Tahap ini dilakukan untuk menyusun daftar fitur dan kebutuhan produk yang akan dikembangkan. Produk Backlog akan berisi daftar tugas yang diprioritaskan berdasarkan apa yang dibutuhkan untuk membangun aplikasi.

b. Sprint Planning

Tahap ini merupakan perencanaan sprint dengan menentukan backlog item yang akan dikerjakan dan disiapkan untuk dikembangkan dalam satu sprint.

c. Sprint Backlog

Tahap ini berisi daftar tugas dari Product Backlog yang dipilih untuk diselesaikan selama sprint. Daftar ini menjadi panduan bagi tim untuk menjalankan pekerjaan yang telah direncanakan, memantau kemajuan, dan menyesuaikan langkah yang diperlukan agar *Sprint Goal* tercapai.

d. Daily Scrum

Tahap ini adalah pertemuan harian yang digunakan untuk menilai kemajuan pekerjaan, mengungkapkan segala hambatan yang diperoleh, dan merencanakan langkah selanjutnya untuk mencapai *Sprint Goal*.

e. Increment

Increment adalah hasil dari sprint yang telah memenuhi *Definition of Done* dan siap digunakan. Setiap sprint dapat menghasilkan lebih dari satu increment yang kemudian ditinjau bersama tim pada Sprint Review.

f. Sprint Review

Tahap ini dilakukan di akhir sprint untuk mengevaluasi increment dan memeriksa Product Backlog yang telah maupun belum diselesaikan. Tahap ini bertujuan untuk menilai hasil pekerjaan, memeriksa kemajuan menuju *Product Goal*, dan menentukan penyesuaian Product Backlog untuk Sprint berikutnya.

g. Sprint Retrospective

Tahap ini dilakukan oleh Scrum Team untuk mengevaluasi proses kerja selama satu sprint dan merencanakan perbaikan untuk sprint berikutnya.

4. Kesimpulan dan Saran

Pada tahap berisikan kesimpulan dari aplikasi pemesanan makanan online di kantin Universitas Pasundan Setiabudi dan juga saran rekomendasi untuk penelitian selanjutnya.

1.6 Sistematika Penulisan

Laporan Tugas Akhir dibuat untuk mendokumentasikan pengerjaan Tugas Akhir. Maka dari itu, diusulkan sistematika penulisan yang menjelaskan mengenai bab-bab pada laporan Tugas Akhir beserta isinya secara rinci. Adapun sistematika penulisannya adalah sebagai berikut :

BAB 1. Pendahuluan

Bab ini berisi gambaran umum mengenai penelitian yang diusulkan dalam penyusunan tugas akhir, yang berisikan mengenai latar belakang, identifikasi masalah, tujuan tugas akhir, lingkup tugas akhir, metodologi tugas akhir, dan sistematika penulisan tugas akhir.

BAB 2. Landasan Teori dan Penelitian Terdahulu

Bab ini berisikan definisi, teori, serta konsep-konsep yang diperlukan dalam pengerjaan tugas akhir. Bab ini juga berisikan hasil penelitian terdahulu yang dijadikan sebagai sumber referensi dalam mengerjakan tugas akhir

BAB 3. Skema Penelitian

Bab ini berisi penjelasan alur penyelesaian tugas akhir, analisis persoalan dan manfaat tugas akhir, kerangka pemikiran teoritis, dan profil tempat penelitian.

BAB 4. Analisis dan Perancangan

Bab ini membahas tahapan awal dalam proses pembangunan yang terdiri dari analisis dan perancangan. Proses analisis melibatkan pendefinisian perangkat lunak dan analisis kebutuhan dengan menggunakan pemodelan spesifikasi kebutuhan.

BAB 5. Implementasi dan Pengujian

Bab ini berisi pembangunan aplikasi berdasarkan hasil analisis dan perancangan yang telah dibuat dan juga pengujian aplikasi yang telah dibuat.

BAB 6. Kesimpulan dan Saran

Pada bab ini berisikan Kesimpulan dan saran dari seluruh proses pembangunan dan hasil penelitian. Bab ini juga memberikan rekomendasi untuk penelitian selanjutnya.

BAB 2

LANDASAN TEORI DAN PENELITIAN TERDAHULU

Bab ini menjelaskan tentang landasan teori dan penelitian terdahulu berisi definisi, teori - teori serta konsep yang digunakan dalam pengerjaan tugas akhir. Dan bab ini membahas tentang penelitian terdahulu yang berkaitan dengan topik pada tugas akhir yang dikerjakan.

2.1 Teori yang Digunakan

Berikut merupakan teori yang digunakan dalam membuat penelitian Tugas Akhir “Rancang Bangun *Front-end* Web Untuk Pemesanan Makanan *Online* Di Universitas Pasundan Setiabudi Menggunakan React JS (Studi Kasus: Kantin Universitas Pasundan Setiabudi)”.

2.1.1 Website

Website adalah kumpulan halaman digital yang berisi informasi seperti teks, animasi, gambar, suara, dan video atau gabungan dari semuanya. Halaman-halaman ini terhubung ke internet, sehingga setiap orang yang terhubung ke internet dapat melihatnya [MUS24].

Sebagai sistem informasi, website menyediakan berbagai layanan dan informasi melalui internet, yang dapat menjadi sumber yang efektif untuk membantu pengguna dalam memperoleh informasi yang mereka butuhkan. Dengan adanya kemudahan akses terhadap layanan yang berguna, seperti pencarian produk, jasa, atau layanan tertentu untuk menjangkau calon pelanggan, website menjadi lebih efisien dan dapat diakses dengan mudah melalui berbagai perangkat yang terhubung ke internet [TJA24].

2.1.2 Responsive Web Design

Responsive Web Design (RWD) adalah metode desain web yang dirancang agar halaman web dapat tampil dengan baik di berbagai perangkat dan ukuran layar, mulai dari yang paling kecil hingga yang terbesar. Pendekatan ini memastikan pengalaman pengguna yang baik dengan menyesuaikan kode HTML dan CSS secara fleksibel sesuai dengan resolusi perangkat, tanpa memerlukan gaya terpisah untuk setiap ukuran layar [GIN22].

2.1.3 Front-end

Front-end adalah segala sesuatu yang menghubungkan pengguna ke *back-end* dan biasanya merupakan *user interface* tempat pengguna berinteraksi dengan sistem. Dengan kata lain, *front-end* adalah terjemahan dari desain UI yang kemudian dimasukkan ke dalam kode program, yang memungkinkan *back-end* untuk melakukan fungsinya dan akhirnya membuat pengguna dapat menggunakannya [FER22].

Bagian ini berperan dalam menampilkan antarmuka pengguna yang dapat dilihat dan digunakan oleh pengguna. Fokus utamanya adalah mengatur antarmuka pengguna serta mengoptimalkan elemen visual agar tampilan lebih menarik dan mudah digunakan. Seorang *front-end developer* bertanggung jawab dalam

merancang dan mengembangkan tata letak (*layout*) pada website maupun aplikasi, dengan tujuan menciptakan pengalaman pengguna yang nyaman dan tidak membosankan. Dalam proses pengembangannya, *front-end* memanfaatkan berbagai teknologi seperti HTML, CSS, dan JavaScript [KUK22].

2.1.4 Javascript

JavaScript merupakan Bahasa pemrograman tingkat tinggi yang digunakan untuk membuat website yang interaktif dan dinamis. JavaScript adalah sebuah bahasa scripting, memiliki kemampuan untuk menambah logika dan fitur pada halaman web tanpa kompilasi. Artinya, dengan menggunakan interpreter, kode javascript dapat dieksekusi langsung tanpa perlu melalui tahap kompilasi terlebih dahulu. Dengan menggunakan interpreter, kode tersebut diubah menjadi instruksi yang dapat dimengerti oleh mesin. Oleh karena itu, bahasa scripting seperti javascript memungkinkan pengembang membuat program yang rumit dengan jumlah kode yang lebih sedikit [ZAH24].

Pada tahun 1995, Netscape Communications Corporation menjadi yang terdepan dalam pengembangan browser web dan web, yang masih merupakan teknologi baru. Pada bulan Mei tahun yang sama, seorang programmer Netscape bernama Brendan Eich membuat JavaScript. Bahasa ini dibuat mudah digunakan, fleksibel, dan sederhana. Saat halaman dimuat, browser akan memproses kode tersebut, memungkinkan tampilan halaman berubah secara dinamis serta merespons interaksi pengguna dengan lebih interaktif.

JavaScript dibuat oleh Netscape dengan tujuan menjadi bahasa pemrograman yang sederhana dan fleksibel untuk web. Awalnya, banyak orang mengira namanya berhubungan dengan Java, tetapi sebenarnya keduanya berbeda. Sejak awal, JavaScript dimaksudkan untuk digunakan di berbagai server dan browser. Namun, pada tahun pertama, muncul masalah ketidakcocokan antara JavaScript Netscape dan versi Microsoft yang direayasa balik. Untuk mengatasi hal ini, Ecma International memulai proses standarisasi pada tahun 1996. Pada tahun 1997, mereka merilis spesifikasi resmi pertama, ECMAScript. Standar ini terus dikembangkan, dan ada beberapa revisi dan penyempurnaan pada akhir 1999. Upaya standarisasi JavaScript dimulai untuk menentukan spesifikasi, melibatkan berbagai kontributor, dan menetapkan proses pengambilan keputusan. Pada tahun 2000, JavaScript sudah banyak digunakan di Web, tetapi dengan menurunnya Netscape dan Brendan Eich berpindah ke proyek lain, kepemimpinan pengembangan JavaScript beralih ke komite standar ECMAScript.

JavaScript mengalami berbagai kegagalan dalam upaya pembaruannya sebelum akhirnya berhasil diperbaiki. Selama bertahun-tahun, komite standar mengalami kebuntuan dan perbedaan pendapat dalam mengembangkan bahasa ini. Namun, pada tahun 2008, tim pengembang berhasil menyelesaikan perbedaan tersebut dan merilis versi ECMAScript yang lebih baik pada tahun 2009.

Keberhasilan ini membuka jalan bagi pengembangan besar selama tujuh tahun, yang akhirnya menghasilkan ECMAScript 2015. Versi ini menjadi landasan utama bagi perkembangan JavaScript modern

dengan banyak peningkatan signifikan. Setelah rilis ECMAScript 2015, proses pembaruan JavaScript diubah agar lebih cepat dan dilakukan secara berkala setiap tahun. Tujuannya adalah untuk terus menyesuaikan bahasa ini dengan kebutuhan pengembang, menambahkan fitur baru, serta meningkatkan efisiensi dan fleksibilitas JavaScript [WIR20].

2.1.5 React JS

React JS merupakan *library* JavaScript yang bersifat *open source* dibuat untuk membangun antarmuka pengguna pada aplikasi web. React JS dikembangkan oleh *software developer* Facebook, Jordan Walke, pada tahun 2011, namun baru tersedia untuk umum pada bulan Mei 2013.

Dengan adanya react js memberikan dampak besar terhadap cara pengembangan aplikasi web. Dalam react, setiap elemen tampilan (komponen) memiliki kemampuan untuk mengelola datanya sendiri, yang dikenal dengan istilah *state*, serta menampilkan antarmuka sesuai dengan data tersebut. React menerapkan konsep komponen, yaitu bagian-bagian kecil dari antarmuka pengguna yang dapat digunakan kembali di berbagai bagian aplikasi. Pendekatan ini menjadikan proses pengembangan lebih terstruktur, efisien, dan mudah dikelola. Selain itu, React juga menghindari manipulasi langsung terhadap struktur HTML (DOM), karena seluruh proses pembaruan tampilan ditangani secara otomatis oleh sistem React [MAU23].

React JS memiliki beberapa keunggulan diantaranya memberikan kecepatan, *simplicity*, dan *scalability*. React JS memungkinkan pengembang dapat membangun sebuah komponen UI yang lebih interaktif, *stateful*, & *reusable*. Di dalam kaidah MVC (*Model View Control*), react js bertanggung jawab pada bagian *view*. Untuk meningkatkan kinerja dan menyederhanakan proses pengembangan, React menggunakan teknologi Virtual DOM yang membuat pembaruan tampilan menjadi lebih cepat dan efisien. Konsep inti dalam react adalah sebagai berikut [HAD24]:

2.1.5.1 Komponen (Components)

Konsep utama dalam React adalah komponen. Komponen merupakan potongan-potongan kecil dari antarmuka pengguna (UI) yang dapat berkembang seiring waktu dan berfungsi sebagai sumber data untuk tampilan (*Views*). Komponen-komponen ini dapat digabungkan untuk membangun antarmuka pengguna aplikasi yang lengkap. Penggunaan komponen memungkinkan pengembang untuk membagi antarmuka pengguna menjadi bagian-bagian kecil yang dapat digunakan kembali dan merancang antarmuka dengan cara yang lebih efisien.

2.1.5.2 JSX (JavaScript XML)

JSX adalah ekstensi sintaks untuk javascript yang digunakan dalam React. Dengan JSX memungkinkan penulisan elemen HTML seperti XML secara langsung di dalam kode JavaScript atau React. Tujuan penggunaan JSX bersama React adalah untuk memudahkan dalam mendeskripsikan tampilan

antarmuka pengguna (UI), serta memungkinkan React memberikan pesan atau peringatan yang lebih jelas dan bermanfaat bagi pengembangan selama proses pengembangan aplikasi.

2.1.5.3 State dan Props

Dalam React JS, pengelolaan data pada komponen sangat penting agar antarmuka pengguna (UI) bersifat dinamis dan responsif. Terdapat dua mekanisme utama yang digunakan, yaitu state dan props, yang memiliki fungsi berbeda namun saling melengkapi dalam komunikasi antar komponen dan pengendalian perubahan data.

a. State

State merupakan data yang dikelola di dalam komponen dan dapat berubah. Perubahan state akan memicu *re-render* komponen untuk mencerminkan perubahan tersebut dalam UI.

b. Props

Props merupakan singkatan dari "*properties*", props adalah cara untuk mengirimkan data dari komponen induk ke komponen anak. Props bersifat *read-only*, artinya komponen anak tidak bisa mengubah nilai props yang diterimanya.

2.1.5.4 Virtual DOM

React JS memanfaatkan Virtual DOM sebagai cara yang efisien untuk memperbarui dan me-render komponen. Virtual DOM merupakan representasi memori dari DOM asli. Ketika terjadi perubahan pada state atau props suatu komponen, react tidak langsung memperbarui DOM di browser, melainkan menerapkan perubahan tersebut terlebih dahulu pada Virtual DOM. Selanjutnya, React akan membandingkan Virtual DOM dengan versi dan hanya menerapkan perubahan minimal ke DOM asli. Pendekatan ini secara signifikan mengurangi manipulasi langsung terhadap DOM, sehingga proses pembaruan tampilan menjadi lebih cepat, efisien, serta mengoptimalkan kinerja aplikasi secara keseluruhan.

2.1.6 Tailwind CSS

Tailwind CSS adalah framework CSS yang mempermudah developer dalam membangun antarmuka pengguna secara cepat dan efisien. Framework ini menggunakan pendekatan *utility-first*, di mana kelas-kelas CSS yang disediakan bersifat utilitas dan dapat langsung diterapkan pada elemen HTML. Dengan cara ini, pengembang tidak perlu menulis kode CSS secara manual untuk mengatur tata letak atau gaya, sehingga proses pengembangan menjadi lebih cepat, fleksibel, dan mudah dipelihara [MUS24].

Tailwind CSS juga merupakan kerangka kerja yang responsif dan sangat mendukung kustomisasi. *Framework* ini memberikan kebebasan penuh kepada pengembang untuk menciptakan desain unik tanpa terikat oleh gaya bawaan seperti pada *framework* lain. Selain itu, Tailwind tidak menyertakan *stylesheet* CSS, yang membuatnya lebih ringan dan mempercepat proses pengembangan.

Dalam beberapa tahun terakhir, Tailwind CSS telah mendapatkan popularitas luar biasa di kalangan desainer dan pengembang web. *Framework* ini dirancang untuk mempercepat proses *prototyping* halaman

web agar sesuai kebutuhan. TailwindCSS bekerja dengan memindai semua file HTML, komponen JavaScript, dan *template* lain untuk nama kelas, menghasilkan gaya yang sesuai, lalu menuliskannya ke file CSS statis [SAL23].

2.1.7 Scrum

Scrum merupakan kerangka kerja yang digunakan untuk mengembangkan produk kompleks dengan strategi yang fleksibel. Dalam penerapannya, Scrum menekankan prinsip transparansi, inspeksi, dan adaptasi yang memungkinkan tim memperoleh umpan balik secara terus-menerus, melakukan perbaikan berkelanjutan, serta meningkatkan kualitas dan produktivitas hasil kerja [SAC16].

Untuk menjalankan prinsip-prinsip scrum secara efektif, Scrum Team dibentuk dengan tanggung jawab yang jelas bagi setiap anggotanya. Tim ini terdiri dari beberapa peran utama, yaitu [GUS20] :

1. Product Owner

Product Owner bertanggung jawab untuk menetapkan spesifikasi serta memaksimalkan nilai bisnis dari aplikasi yang dikembangkan. Peran ini memastikan bahwa prioritas pekerjaan dan tujuan produk selalu selaras dengan kebutuhan bisnis dan pemangku kepentingan.

2. Development Team

Development Team bertugas melaksanakan backlog yang telah disusun oleh Product Owner. Setiap anggota memahami tanggung jawabnya, mengikuti *Definition of Done* untuk menjamin kualitas, serta menyesuaikan rencana secara berkala guna mencapai target Sprint.

3. Scrum Master

Scrum Master berperan membimbing dan mendukung tim dalam penerapan Scrum. Tugasnya meliputi memastikan seluruh anggota tim menjalankan Scrum dengan benar, menghilangkan hambatan yang mengganggu progres, dan membantu tim serta organisasi memahami serta menerapkan praktik Scrum secara efektif.

2.1.8 Kantin

Kantin merupakan tempat yang menyediakan makanan dan memungkinkan terjadinya transaksi antara pembeli dan penjual. Pengelolaan kantin merupakan salah satu bisnis makanan yang mempunyai potensi keuntungan yang sangat besar. Dengan pengelolaannya, Teknologi informasi dan komunikasi dapat menjadi bagian dari operasi tersebut [SAZ24].

Selain untuk keperluan bisnis, kantin juga memiliki tujuan lain. Misalnya, di lingkungan kampus, kantin tidak hanya berfungsi sebagai tempat makan, tetapi juga bisa menjadi ruang untuk mengerjakan tugas, berdiskusi, atau sekadar berkumpul bersama teman.

2.1.9 Pemesanan

Pemesanan adalah suatu kegiatan yang dilakukan oleh konsumen sebelum membeli barang. Pemesanan merupakan proses penyampaian informasi yang jelas mengenai suatu item, termasuk identitas pemesanan, jenis barang yang dipesan, serta jumlahnya [ANG23].

Menurut Gouzali (1996 : 31) : “Pemesanan adalah penerimaan pesanan dari pelanggan terhadap suatu produk. Lanjutan dari pemesanan adalah pengiriman produk sampai ketangan pemesan dengan selamat” [SAT19].

2.2 Penelitian Terdahulu

Berikut merupakan daftar penelitian terdahulu yang mendukung dan memiliki kemiripan dengan tugas akhir ini.

Tabel 2. 1 Penelitian Terdahulu

No	Peneliti	Hasil Penelitian	Alasan Digunakan
1.	Luki Rizki, Yuyun Tresnawati [RIZ24]	Hasil dari penelitian ini adalah aplikasi yang dapat memudahkan pelanggan pada saat melakukan pemesanan makanan dan minuman serta memberikan informasi terkait menu dan pelayan dan pemilik kafe dalam mengelola pemesanan.	Penelitian ini digunakan untuk memahami kebutuhan pengguna, perancangan sistem, dan implementasi sistem pemesanan makanan berbasis web.
2.	Dea Syah Kania, Dina Marwah Afrahmi, Dwindi Putri Nuria, Dadang Yusup [KAN23]	Hasil penelitian ini berupa aplikasi E(at)veryday berbasis website yang memungkinkan pengguna untuk melakukan transaksi memesan makanan secara online dengan efisiensi dan kenyamanan yang lebih tinggi. Pengujian yang dilakukan menunjukkan hasil bahwa sistem ini berfungsi dengan baik dan memenuhi tujuan awal pengembangannya.	Penelitian ini digunakan untuk memahami sistem pemesanan makanan secara online berbasis website.
3.	Farida Nurlaila [NUR19]	Hasil dari penelitian ini menunjukkan bahwa aplikasi yang dihasilkan dapat membantu Restoran 1953 Indonesia dalam hal mempromosikan dan menangani pesanan pelanggan melalui fitur reservasi menu dan reservasi tempat, proses menghitung total pembayaran jadi lebih mudah karena prosesnya dilakukan melalui aplikasi dengan mengkalkulasi seluruh pesanan pelanggan.	Penelitian ini digunakan untuk memahami proses pemesanan dan perhitungan total pembayaran secara lebih efektif.
4.	Ignatius Daniel Y.T.A.Ginting, Herman Tolle, Komang Candra Brata [GIN22]	Penelitian ini menghasilkan aplikasi manajemen sampah berbasis web dengan fitur tambah dan melihat data sampah untuk memudahkan pengolahan data yang dilakukan oleh Dinas Lingkungan Hidup pada TPS3R Kota Batu.	Penelitian ini digunakan untuk memahami penerapan React JS dan Tailwind CSS dalam membangun UI yang responsif dan interaktif, sehingga tampilan aplikasi dapat menyesuaikan dengan berbagai ukuran layar dan perangkat untuk meningkatkan pengalaman pengguna.
5.	Kevin Juan, Setia Budi [JUA23]	Penelitian ini menghasilkan aplikasi menu digital yang responsif dan efisien.	Penelitian ini digunakan untuk memberikan pemahaman tentang bagaimana React JS digunakan untuk membangun UI yang dinamis, efisien, dan modular dalam sistem menu digital. Dengan konsep komponen yang dapat digunakan kembali serta pengelolaan state yang lebih optimal, penelitian ini menjadi referensi penting dalam merancang tampilan dan interaksi yang lebih baik dalam sistem pemesanan makanan online.

BAB 3

SKEMA PENELITIAN

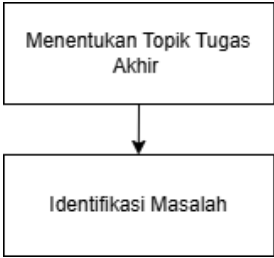
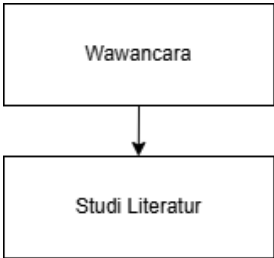
Bab ini berisi penjelasan mengenai uraian tentang alur pengerjaan tugas akhir, analisis masalah, solusi yang diberikan dan profil tempat penelitian.

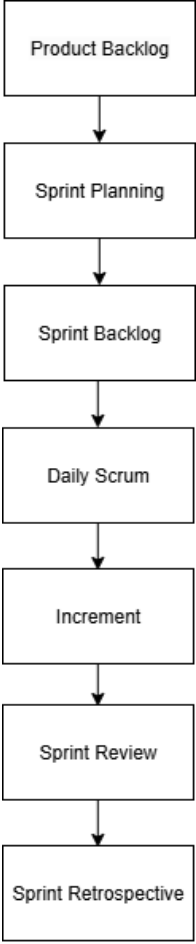
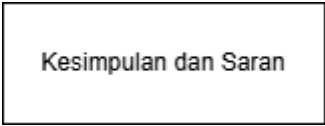
3.1 Alur Pengerjaan Tugas Akhir

Alur penelitian merupakan kerangka pengerjaan tugas akhir yang menjelaskan mengenai tahapan dan langkah-langkah yang spesifik dalam pengerjaan tugas akhir. Pada konteks pengerjaan tugas akhir ini, alur pengerjaan dibagi menjadi tiga bagian utama, yaitu:

1. Tahap dan Hasil, yaitu berisi gambaran serta tahapan proses yang akan dikerjakan dalam tugas akhir.
2. Langkah penelitian, yaitu berisi urutan langkah yang dilakukan dalam pengerjaan tugas akhir.
3. Referensi, yaitu berisi sumber pengetahuan yang dijadikan landasan teori dalam mengerjakan tugas akhir.

Tabel 3. 1 Alur Pengerjaan Tugas Akhir

Tahap & Hasil	Langkah Penelitian	Referensi
Tahap 1: Mentukan topik TA dan identifikasi masalah Hasil: Mendapatkan topik TA dan masalah yang telah teridentifikasi Kontribusi: Memberikan batasan dalam analisis penelitian serta tahap pengumpulan data	 <pre> graph TD A[Menentukan Topik Tugas Akhir] --> B[Identifikasi Masalah] </pre>	
Tahap 2: Melakukan pengumpulan data dan studi literatur yang mendukung topik tugas akhir Hasil: teori yang relevan dan mendapatkan data yang dikumpulkan dari hasil wawancara penjual dan pelanggan kantin Unpas Kontribusi: Berguna untuk tahap selanjutnya yaitu pada tahapan perancangan	 <pre> graph TD A[Wawancara] --> B[Studi Literatur] </pre>	1. Wawancara dengan penjual dana pelanggan kantin Unpas 2. Kantin [SAZ24]. 3. Pemesanan [ANG23] [SAT19]

Tahap & Hasil	Langkah Penelitian	Referensi
Tahap 3 : Melakukan perancangan dan pembangunan aplikasi dengan metode scrum Hasil : Menghasilkan tampilan <i>front-end</i> website pemesanan makanan online	 <pre> graph TD A[Product Backlog] --> B[Sprint Planning] B --> C[Sprint Backlog] C --> D[Daily Scrum] D --> E[Increment] E --> F[Sprint Review] F --> G[Sprint Retrospective] </pre>	1. Scrum [GUS20] [SCH20] [SAC16] 2. React JS [MAU23] [HAD24] 3. Tailwind CSS [MUS24] [SAL23] 4. Perancangan dan Pengembangan Aplikasi E(AT)VERYDAY sebagai Sistem Pemesanan Kantin Berbasis Website [KAN23] 5. Rancang Bangun Aplikasi Pelayanan Pemesanan Makanan Dan Minuman Berbasis Website [RIZ24]
Tahap 4 : Membuat Kesimpulan dan Saran Hasil : Kesimpulan dan Saran tugas akhir Kontribusi : Penyimpulan dan saran untuk penelitian berikutnya	 <pre> graph TD A[Kesimpulan dan Saran] </pre>	

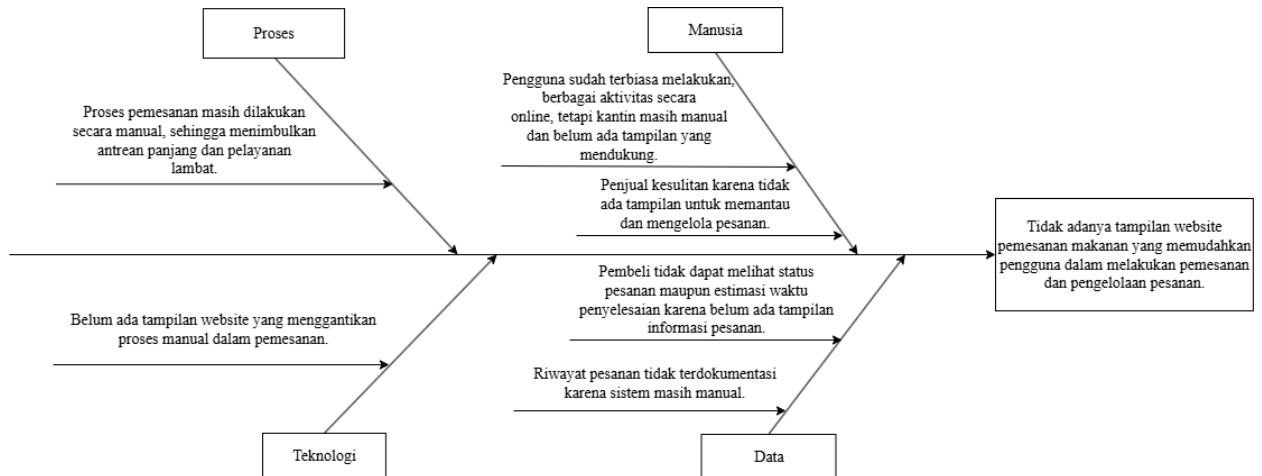
3.2 Perumusan Masalah

Perumusan masalah memiliki tujuan untuk mengetahui kesesuaian solusi terhadap permasalahan yang ada pada penelitian tugas akhir. Perumusan masalah dijelaskan ke dalam dua subbab yaitu analisi sebab akibat dan analisi solusi masalah.

3.2.1 Analisis Sebab Akibat

Analisis dari perancangan sistem informasi pemesanan makanan online berbasis web di kantin ini bertujuan untuk mengetahui solusi permasalahna dari tugas akhir ini. Sehingga dapat menemukan

permasalahan secara detail terhadap permasalahan yang ada di dalam prose pemesanan makanan di kantin dengan solusi yang tepat. Tahap analisis pada tahap ini menggunakan diagram sebab akibat diagram ini disebut dengan Tulang Ikan Diagram (Fishbone Diagram). Berikut ini merupakan fishbone diagram analisis persoalan arsitektur informasi. Dapat dilihat pada gambar berikut:



Gambar 3. 1 Fishbone Sebab Akibat

Berikut merupakan deskripsi dari persoalan yang terjadi:

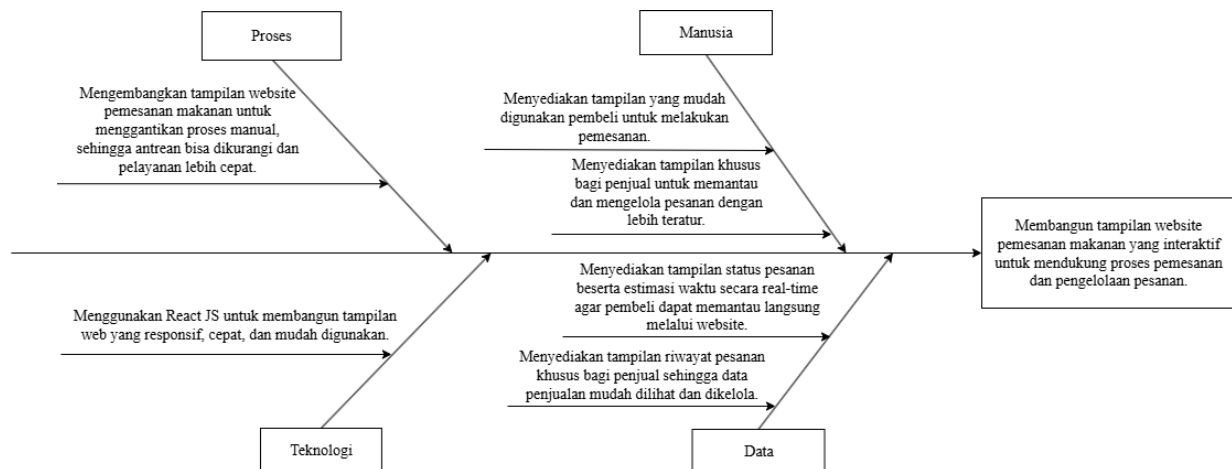
Tabel 3. 2 Detail Fishbone Sebab Akibat

No	Komponen Utama	Komponen Detail	Penjelasan
1	Proses	Proses pemesanan masih dilakukan secara manual, sehingga menimbulkan antrean panjang dan pelayanan lambat.	Saat ini, pemesanan makanan di kantin masih dilakukan secara manual tanpa adanya tampilan website yang mempermudah proses. Pembeli harus datang langsung ke lokasi untuk memesan, kemudian menunggu pesanan diproses secara bergantian. Kondisi ini sering menimbulkan antrean panjang, terutama pada jam sibuk, sehingga waktu tunggu menjadi lama. Selain memperlambat pelayanan, metode manual ini juga rentan menimbulkan kesalahan dalam pencatatan pesanan serta menurunkan kepuasan pembeli.
2	Teknologi	Belum ada tampilan website yang menggantikan proses manual dalam pemesanan.	Kantin saat ini belum memiliki tampilan website yang dapat digunakan pembeli untuk melakukan pemesanan secara digital. Seluruh proses masih dilakukan secara konvensional dengan datang langsung ke kios penjual. Kondisi ini membuat pemesanan menjadi kurang efisien karena pembeli harus meluangkan waktu untuk antre. Tidak adanya sistem berbasis website juga membatasi akses informasi terkait menu, harga, maupun status pesanan, sehingga pengalaman pembeli menjadi kurang praktis dan modern.
3	Manusia	Pengguna sudah terbiasa melakukan berbagai aktivitas secara online, tetapi kantin masih manual dan belum ada tampilan yang mendukung.	Saat ini, sebagian besar pengguna sudah terbiasa melakukan aktivitas secara online, mulai dari berbelanja, memesan makanan, hingga melakukan transaksi keuangan. Namun, kantin masih menerapkan sistem manual tanpa adanya tampilan website yang mendukung proses pemesanan. Hal ini menimbulkan ketidaksesuaian dengan kebutuhan pengguna yang serba digital, sehingga pengalaman pemesanan menjadi kurang praktis dan efisien.
		Penjual kesulitan karena tidak ada tampilan untuk memantau dan mengelola pesanan.	Saat ini penjual tidak memiliki tampilan yang dapat digunakan untuk memantau maupun mengelola pesanan. Semua proses dilakukan secara manual, mulai dari mencatat pesanan hingga memastikan pesanan sudah selesai. Kondisi ini menyulitkan penjual dalam mengatur alur pesanan, meningkatkan risiko

No	Komponen Utama	Komponen Detail	Penjelasan
			kesalahan, serta membuat proses pelayanan menjadi kurang efisien.
4	Data	Pembeli tidak dapat melihat status pesanan maupun estimasi waktu penyelesaian karena belum ada tampilan informasi pesanan.	Tidak tersedianya tampilan informasi pesanan membuat pembeli tidak mengetahui perkembangan pesanan mereka. Pembeli hanya bisa menunggu tanpa kepastian mengenai status maupun estimasi waktu penyelesaian, sehingga menimbulkan ketidaknyamanan dan berpotensi menurunkan kepuasan.
		Riwayat pesanan tidak terdokumentasi karena sistem masih manual.	Pada proses manual, penjual tidak memiliki tampilan untuk menyimpan dan melihat riwayat pesanan. Hal ini menyulitkan penjual dalam memantau hasil penjualan maupun melakukan evaluasi.

3.2.2 Analisis Solusi Masalah

Analisis solusi masalah dilakukan dengan menggunakan diagram fishbone sebagai solusi yang muncul dari masalah yang sedang berlangsung. Berikut merupakan analisis solusi masalah yang terdapat pada penelitian tugas akhir.



Gambar 3. 2 Fishbone Solusi Masalah

Berikut merupakan penjelasan mengenai detail hasil dari analisis solusi dari masalah yang ada digambarkan pada diagram diatas:

Tabel 3. 3 Detail Fishbone Solusi Masalah

No	Komponen Utama	Komponen Detail	Penjelasan
1	Proses	Mengembangkan tampilan website pemesanan makanan untuk menggantikan proses manual, sehingga antrean bisa dikurangi dan pelayanan lebih cepat.	Dengan adanya tampilan website, pembeli dapat melakukan pemesanan secara langsung melalui sistem tanpa harus mengantri di kios. Hal ini akan mempercepat proses pemesanan, meminimalisir kesalahan pencatatan, serta meningkatkan kepuasan pembeli karena proses menjadi lebih praktis dan efisien.
2	Teknologi	Menggunakan React JS untuk membangun tampilan web yang responsif, cepat, dan mudah digunakan.	React JS dipilih karena mampu menghasilkan tampilan web yang interaktif dan efisien. Dengan React, sistem pemesanan dapat berjalan lebih cepat, mudah diakses dari berbagai perangkat, serta memberikan pengalaman pengguna yang lebih baik dibandingkan metode manual.
3	Manusia	Menyediakan tampilan yang mudah digunakan pembeli untuk melakukan pemesanan.	Dengan adanya tampilan khusus bagi pembeli, proses pemesanan menjadi lebih praktis tanpa harus antri langsung di kantin. Tampilan yang sederhana dan user-friendly

No	Komponen Utama	Komponen Detail	Penjelasan
			memudahkan pembeli memilih menu, menambahkan pesanan, serta melakukan konfirmasi dengan cepat.
		Menyediakan tampilan khusus bagi penjual untuk memantau dan mengelola pesanan dengan lebih teratur.	Dengan adanya tampilan yang dirancang khusus untuk penjual, setiap pesanan dapat dipantau secara real-time, sehingga proses pengelolaan menjadi lebih mudah dan terorganisir. Hal ini membantu penjual mengurangi risiko kesalahan pencatatan, mempercepat pelayanan, serta memudahkan dalam mengatur alur pesanan yang masuk.
4	Data	Menyediakan tampilan status pesanan beserta estimasi waktu secara real-time agar pembeli dapat memantau langsung melalui website.	Dengan adanya tampilan status pesanan, pembeli dapat mengetahui sejauh mana proses pesannya sudah berjalan, mulai dari diterima, diproses, hingga selesai. Estimasi waktu yang ditampilkan secara real-time membantu mengurangi ketidakpastian, meningkatkan kenyamanan, serta meminimalisir kebutuhan pembeli untuk terus menanyakan status pesannya kepada penjual.
		Menyediakan tampilan riwayat pesanan khusus bagi penjual sehingga data penjualan mudah dilihat dan dikelola.	Dengan adanya tampilan riwayat pesanan, penjual dapat melihat kembali transaksi yang sudah terjadi secara lebih terstruktur. Hal ini memudahkan penjual dalam melakukan evaluasi penjualan, pencatatan data, serta mengelola laporan dengan lebih rapi dibandingkan proses manual yang sering kali tidak terdokumentasi.

3.3 Profil Penelitian Tugas Akhir



Gambar 3. 3 Logo Unpas

Nama Tempat : Kantin Universitas Pasundan Setiabudi
 Alamat : Jl. Dr. Setiabudi No.193, Gegerkalong, Kec.
 Sukasari, Kota Bandung, Jawa Barat 40153

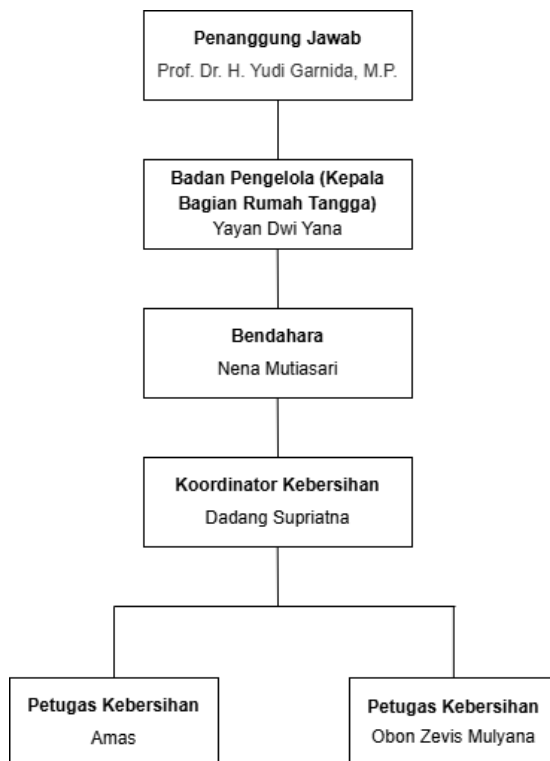
3.3.1 Deskripsi Singkat

Sebelum mengalami perubahan besar, kantin UNPAS masih dalam kondisi kurang terawat dan dikelola oleh Koperasi Warga UNPAS. Namun, seiring berjalannya waktu, pihak universitas mulai memberikan perhatian lebih terhadap kondisi ini. Wakil Dekan II kemudian mengambil langkah untuk mengalihkan pengelolaan kantin langsung ke universitas. Sebagai bagian dari upaya perbaikan, dilakukan renovasi bangunan agar lebih nyaman dan layak digunakan.

Saat ini, kantin UNPAS memiliki sekitar 50 kios, dengan skema penyewaan yang mengutamakan pedagang lama yang telah berjualan sebelum renovasi, berjumlah sekitar 25 orang. Prioritas berikutnya

diberikan kepada pensiunan staf dan dosen UNPAS, serta masyarakat sekitar kampus sebagai bentuk tanggung jawab sosial universitas. Dengan sistem pengelolaan yang lebih terstruktur, kantin UNPAS kini menjadi fasilitas yang lebih nyaman, tertata, dan mendukung kebutuhan mahasiswa, dosen, dan staf serta warga sekitar.

3.3.2 Struktur Organisasi



Gambar 3. 4 Struktur Organisasi Kantin Unpas

BAB 4

ANALISIS DAN PERANCANGAN

Bab ini membahas proses awal yang dilakukan sebelum pembangunan desain dan pembangunan aplikasi pemesanan makanan dan minuman digital di kantin Universitas Pasundan Setiabudi.

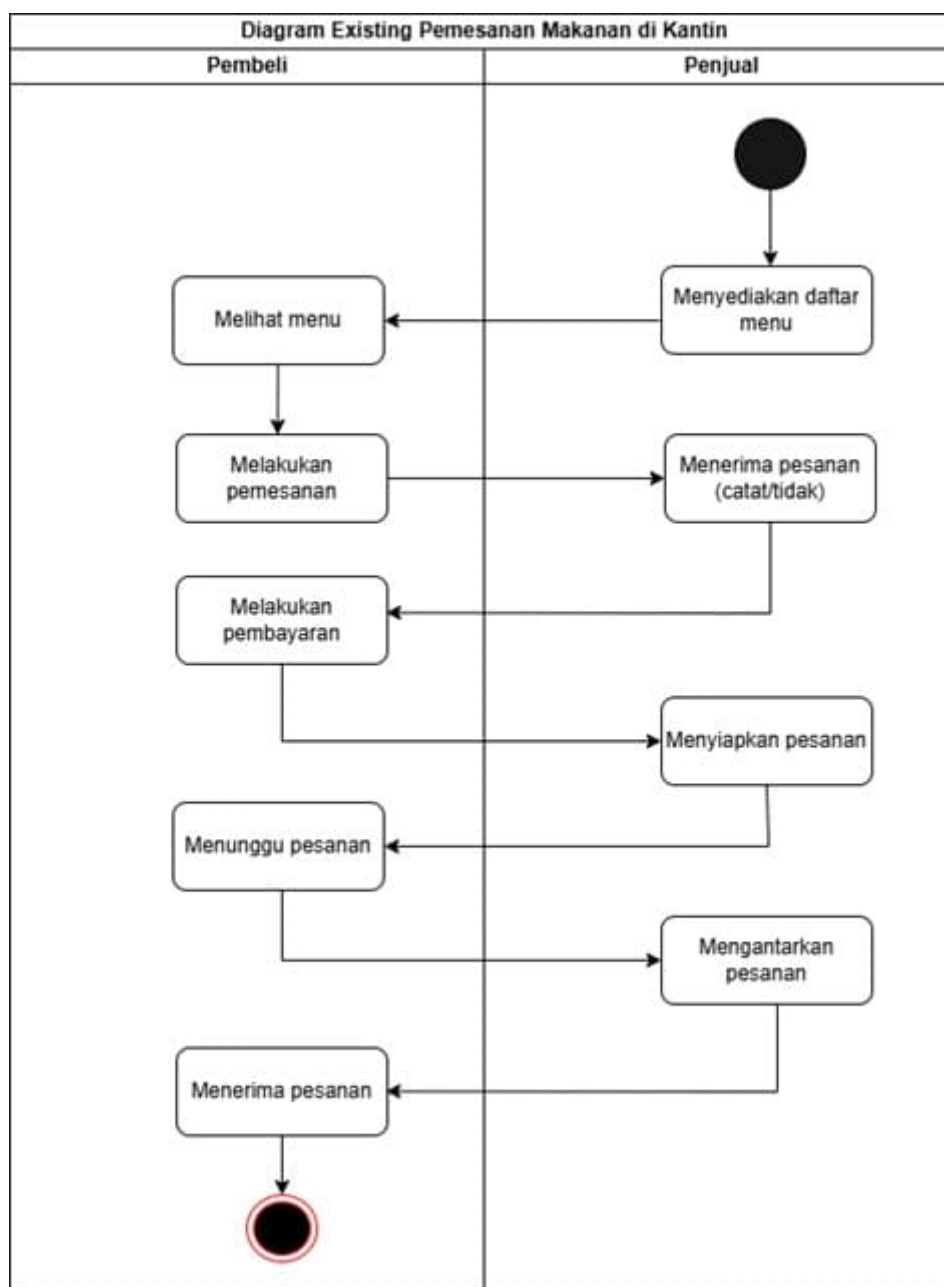
4.1 Analisis Kebutuhan Perangkat Lunak

Analisis kebutuhan perangkat lunak bertujuan untuk mengidentifikasi dan memahami kebutuhan sistem yang akan dikembangkan, termasuk spesifikasi, karakteristik, dan fitur yang akan dibangun. Proses ini memastikan bahwa perangkat lunak yang dikembangkan sesuai dengan tujuan serta kebutuhan pengguna.

4.1.1 Analisis Sistem Saat Ini

Analisis sistem saat ini merupakan tahap untuk menganalisis dan menilai sistem yang telah digunakan sebelum merancang perangkat lunak. Tujuan utamanya adalah memahami kelebihan, kekurangan, peluang pengembangan, serta potensi hambatan dari sistem yang sedang berjalan. Hasil dari analisis ini akan menjadi dasar dalam merancang desain dan menentukan fitur yang sesuai untuk sistem yang akan dibangun.

Tabel 4. 1 Diagram *Swimlane* Analisis Sistem Saat Ini
Diagram *Swimlane* Analisis Sistem Saat Ini



Gambar 4. 1 Diagram *Swimlane* Sistem Saat Ini

Deskripsi

Diagram swimlane ini menjelaskan aktivitas yang dilakukan oleh pembeli dan penjual dalam proses pemesanan makanan di kantin pada sistem yang saat ini masih berjalan secara manual.

4.1.2 Analisis Kebutuhan Pengguna

Analisis kebutuhan pengguna adalah proses menentukan dan memahami kebutuhan, harapan, dan kendala pengguna yang akan menggunakan perangkat lunak, yang dapat mempengaruhi pembangunan fitur dari aplikasi yang akan dibangun.

Tabel 4. 2 *User Requirement*

Kode	User Requirement
UR-01	Penjual ingin dapat menyediakan data menu untuk pembeli
UR-02	Pembeli ingin dapat melihat daftar menu dengan jelas
UR-03	Pembeli ingin dapat melakukan pemesanan dengan mudah
UR-04	Penjual ingin dapat menerima notifikasi saat ada pesanan masuk
UR-05	Pembeli ingin dapat melakukan pembayaran dengan mudah
UR-06	Penjual ingin dapat menyiapkan pesanan sesuai pesanan
UR-07	Pembeli ingin dapat menunggu pesanan dengan informasi yang jelas
UR-08	Penjual ingin dapat menyerahkan pesanan kepada pembeli
UR-09	Pembeli ingin dapat menerima pesanan sesuai pesannya

4.1.3 Analisis Kebutuhan Fungsional

Analisis kebutuhan fungsional adalah proses untuk mengidentifikasi dan mendefinisikan fitur atau fungsi tertentu yang harus dimiliki oleh sebuah sistem atau aplikasi agar dapat memenuhi kebutuhan pengguna. Berikut merupakan tabel kebutuhan fungsional.

Tabel 4. 3 *Software Requirement*

Kode	User Requirement	Kode	Software Requirement
UR-01	Penjual ingin dapat menyediakan data menu untuk pembeli.	KF-01	Perangkat lunak harus dapat mengelola menu.
UR-02	Pembeli ingin dapat melihat daftar menu dengan jelas.	KF-02	Perangkat lunak harus dapat menampilkan daftar kios.
		KF-03	Perangkat lunak harus dapat menampilkan daftar menu perkios.
		KF-04	Perangkat lunak harus dapat menampilkan detail menu.
		KF-05	Perangkat lunak harus dapat mencari menu berdasarkan <i>keyword</i> .
UR-03	Pembeli ingin dapat melakukan pemesanan dengan mudah.	KF-06	Perangkat lunak harus dapat mengelola keranjang
		KF-07	Perangkat lunak harus dapat membuat pesanan dari keranjang dengan data pembeli yang lengkap.
UR-04	Penjual ingin dapat menerima notifikasi saat ada pesanan masuk.	KF-08	Perangkat lunak harus dapat mengirim notifikasi saat ada pesanan masuk.
UR-05	Pembeli ingin dapat melakukan pembayaran dengan mudah.	KF-09	Perangkat lunak harus dapat menyediakan pembayaran menggunakan <i>payment gateway</i>
UR-06	Penjual ingin dapat menyiapkan pesanan sesuai pesanan	KF-10	Perangkat lunak harus dapat menampilkan daftar pesanan masuk beserta detail pesanan.
		KF-11	Perangkat lunak harus dapat mengubah status pesanan.
UR-07	Pembeli ingin dapat menunggu pesanan dengan informasi yang jelas.	KF-12	Perangkat lunak harus dapat menampilkan status pesanan dan estimasi waktu selesai pesanan.
UR-08	Penjual ingin dapat menyerahkan pesanan kepada pembeli.	KF-13	Perangkat lunak harus dapat menampilkan tipe pengantaran sesuai dengan yang dipilih pembeli.
UR-09	Pembeli ingin dapat menerima pesanan sesuai pesannya.	KF-14	Perangkat lunak harus dapat mengirim notifikasi kepada pembeli ketika pesanan selesai.

4.1.4 Analisis Kebutuhan Non Fungsional

Analisis kebutuhan non-fungsional adalah proses yang dilakukan untuk mengidentifikasi dan mendefinisikan kebutuhan atau persyaratan yang tidak berkaitan langsung dengan fungsi atau layanan sistem, tetapi lebih kepada bagaimana cara sistem berfungsi serta karakteristik apa yang harus dimiliki. Berikut merupakan tabel kebutuhan non fungsional.

Tabel 4. 4 Analisis Kebutuhan Non Fungsional

Kode	Kebutuhan Non Fungsional
KNF-01	Perangkat lunak harus dapat mengelola autentikasi pengguna

4.2 Pemodelan Berbasis Skenario

Pemodelan berbasis skenario digunakan untuk menggambarkan interaksi antara aktor dan sistem yang akan dikembangkan. Proses ini biasanya dimulai dengan pembuatan Use Case serta diagram aktivitas, sehingga dapat memberikan pemahaman yang lebih jelas mengenai kebutuhan dan perilaku sistem.

4.2.1 Deskripsi Aktor

Aktor adalah pengguna atau entitas yang berinteraksi dengan sistem. Bagian ini menjelaskan aktor dalam *Use Case Diagram*. Berikut adalah tabel definisi aktor.

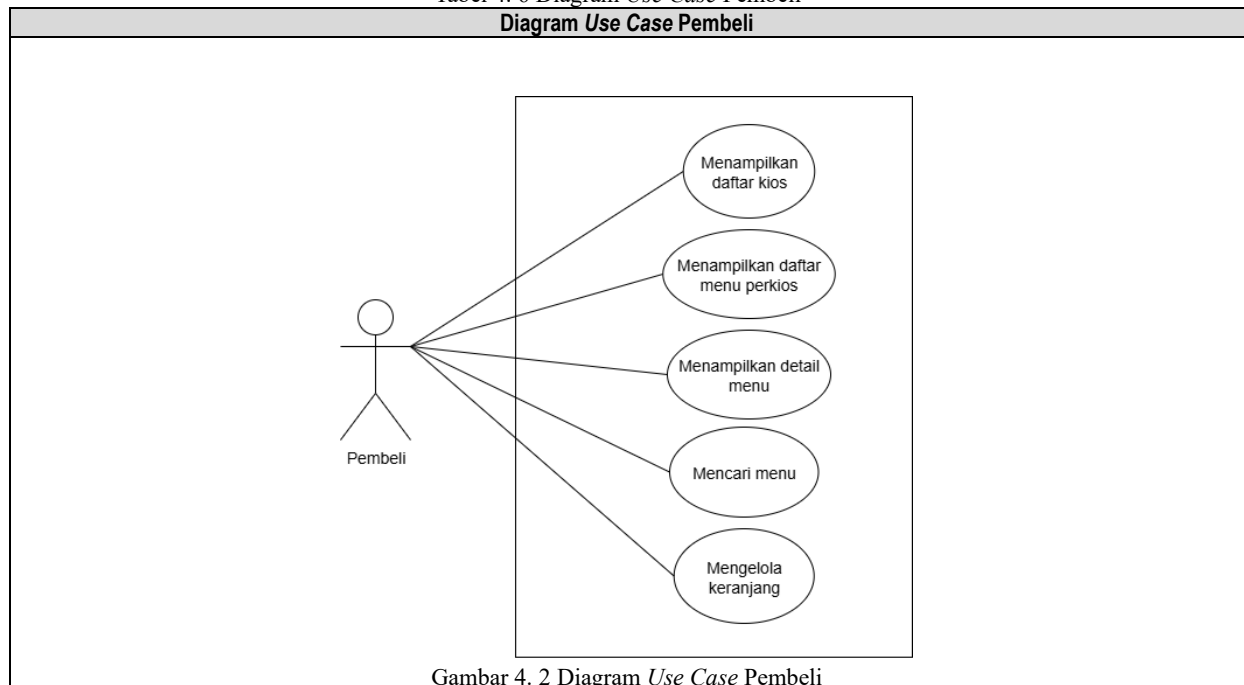
Tabel 4. 5 Deskripsi Aktor

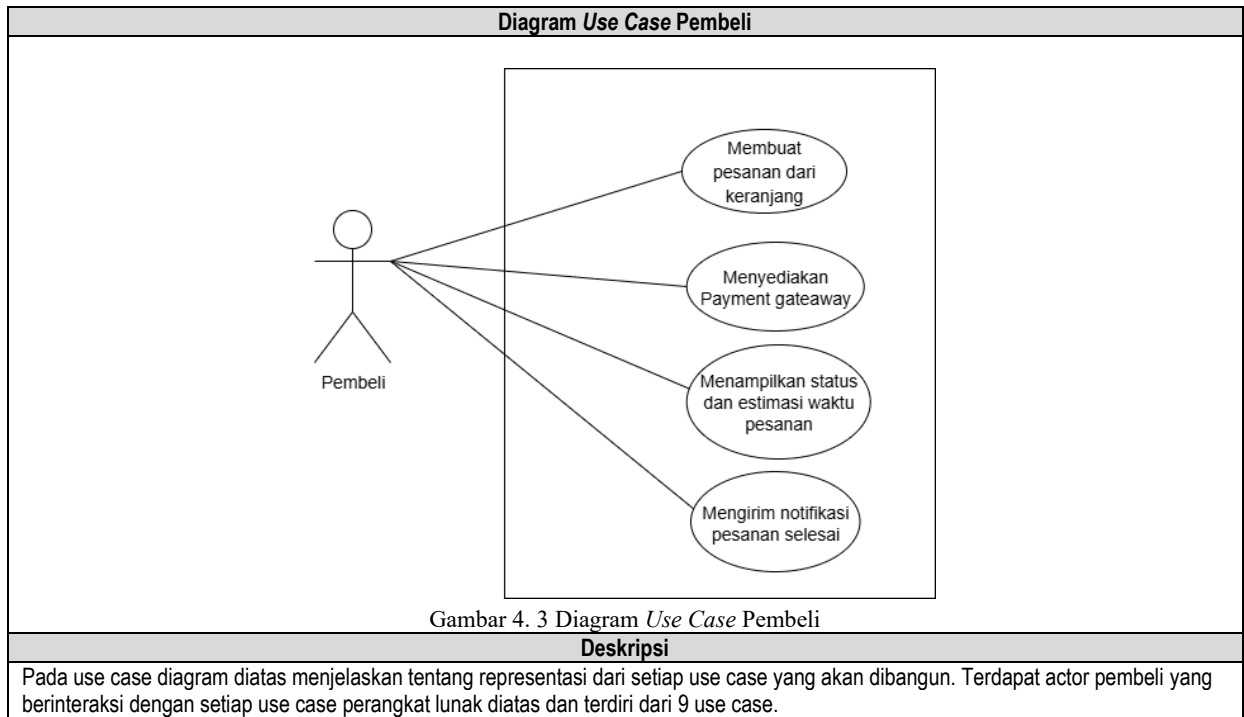
No	Aktor	Deskripsi Aktor
1	Pembeli	Orang yang ingin memesan makanan dan minuman dari kios kantin melalui website
2	Penjual	Orang yang memiliki kios dan bertanggung jawab mengelola menu serta menerima dan memproses pesanan dari pembeli melalui website

4.2.2 Diagram Use Case

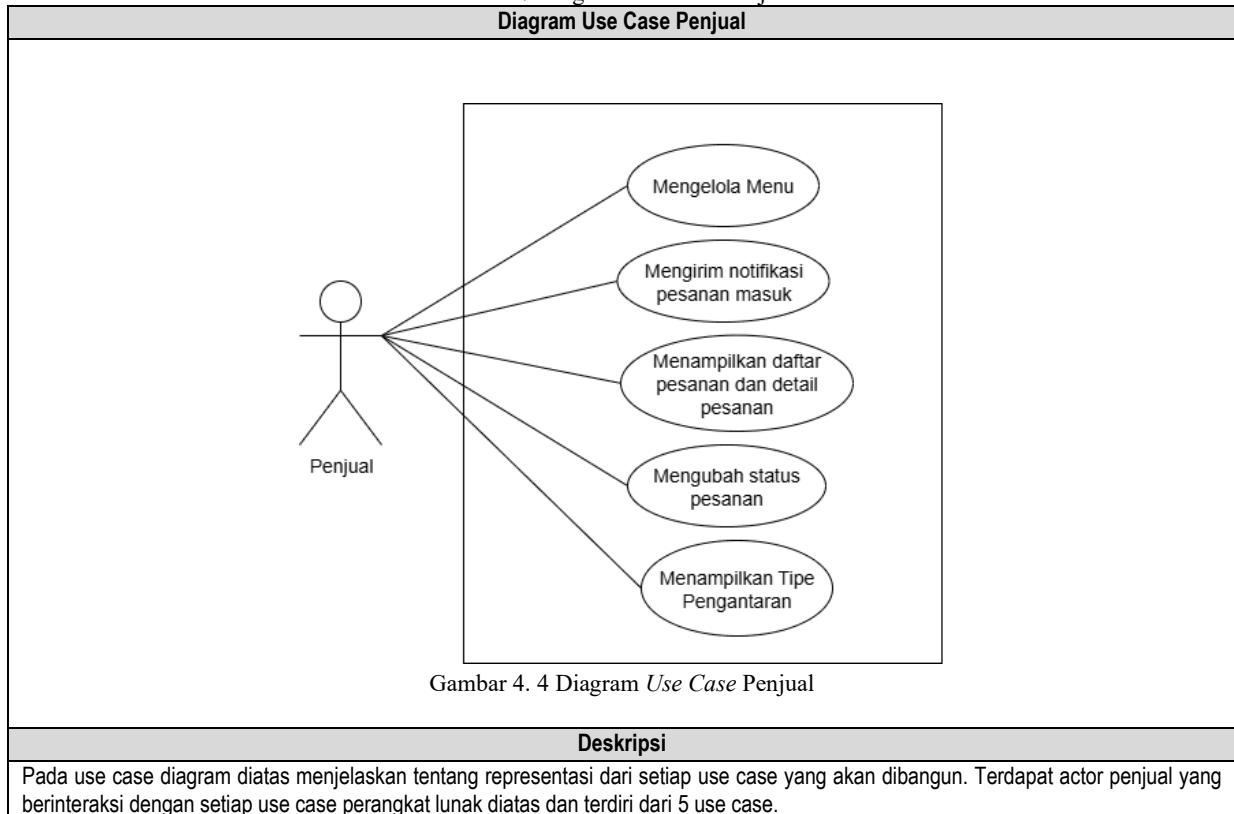
Diagram *Use Case* adalah gambaran tentang bagaimana pengguna atau aktor berinteraksi dengan sistem, yang menggambarkan fungsi atau layanan yang disediakan oleh sistem dalam berbagai skenario penggunaan. Berikut merupakan diagram *Use Case*.

Tabel 4. 6 Diagram *Use Case* Pembeli





Tabel 4. 7 Diagram Use Case Penjual



4.2.3 Deskripsi Diagram Use Case

Berikut merupakan deskripsi detail dari use case yang ditampilkan pada gambar di atas, yang akan dijelaskan pada tabel di bawah ini:

Tabel 4. 8 Deskripsi Diagram Use Case Pembeli

Kode	Nama Use Case	Deskripsi
UC-01	Menampilkan daftar kios	Use case ini digunakan untuk menampilkan daftar kios yang tersedia agar pembeli dapat memilih kios tujuan.
UC-02	Menampilkan daftar menu perkios	Use case ini digunakan untuk menampilkan daftar menu dari kios yang dipilih.
UC-03	Menampilkan detail menu	Use case ini digunakan untuk menampilkan informasi detail dari menu yang dipilih, seperti foto, nama menu, harga, dan deskripsi, sehingga pembeli dapat menentukan pilihan dengan lebih jelas.
UC-04	Mencari menu	Use case ini digunakan untuk memudahkan pembeli menemukan menu tertentu dengan menggunakan kata kunci pencarian.
UC-05	Mengelola keranjang	Use case ini digunakan untuk menambahkan, memperbarui, atau menghapus menu yang ada di dalam keranjang sebelum pesanan dikonfirmasi.
UC-06	Membuat pesanan dari keranjang	Use case ini digunakan untuk membuat pesanan berdasarkan isi keranjang dengan melengkapi data pembeli.
UC-07	Menyediakan <i>Payment gateway</i>	Use case ini digunakan untuk memfasilitasi proses pembayaran.
UC-08	Menampilkan status dan estimasi waktu pesanan	Use case ini digunakan untuk menampilkan status pesanan yang sedang diproses serta estimasi waktu selesai agar pembeli mendapatkan informasi yang jelas.
UC-09	Mengirim notifikasi pesanan selesai	Use case ini digunakan untuk memberikan notifikasi kepada pembeli ketika pesanan telah selesai.

Tabel 4. 9 Deskripsi Diagram Use Case Penjual

Kode	Nama Use Case	Deskripsi
UC-10	Mengelola Menu	Use case ini digunakan untuk menambahkan, mengedit, dan menghapus menu agar informasi yang ditampilkan ke pembeli selalu sesuai.
UC-11	Mengirim notifikasi pesanan masuk	Use case ini digunakan untuk memberikan pemberitahuan kepada penjual saat ada pesanan baru dari pembeli.
UC-12	Menampilkan daftar pesanan dan detail pesanan	Use case ini digunakan untuk menampilkan daftar pesanan yang masuk beserta detailnya, seperti menu, jumlah, catatan pembeli, dan total harga.
UC-13	Mengubah status pesanan	Use case ini digunakan untuk memperbarui status pesanan (menunggu diproses, sedang diproses, dan selesai) agar pembeli mengetahui update status pesannya.
UC-14	Menampilkan tipe pengantaran	Use case ini digunakan untuk menampilkan pilihan tipe pengantaran pesanan (ambil sendiri atau antar) sesuai dengan yang dipilih pembeli.

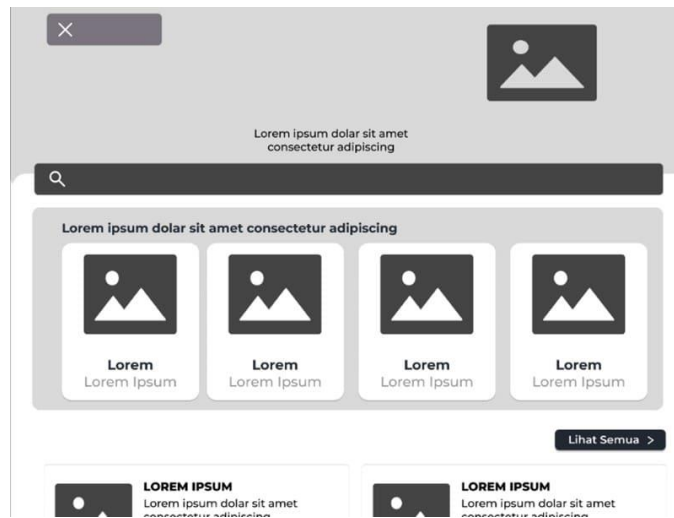
4.2.4 Skenario Use Case

Skenario *use case* merupakan deskripsi alur penggunaan aplikasi yang dijalankan sesuai dengan *use case* yang sudah dirancang sebelumnya. Pada penelitian ini, skenario *use case* menggambarkan bagaimana interaksi pengguna dengan aplikasi pemesanan makanan *online* di Kantin Universitas Pasundan Setiabudi. Rincian skenario tersebut ditampilkan pada beberapa tabel berikut.

Tabel 4. 10 Skenario UC-01

Identifikasi	
Kode	UC-01
Nama	Menampilkan daftar kios
Tujuan	Pembeli melihat daftar kios
Deskripsi	Pembeli mengakses halaman beranda, lalu memilih menu lihat semua untuk melihat kios-kios apa saja yang ada
Aktor	Pembeli
Skenario Utama	

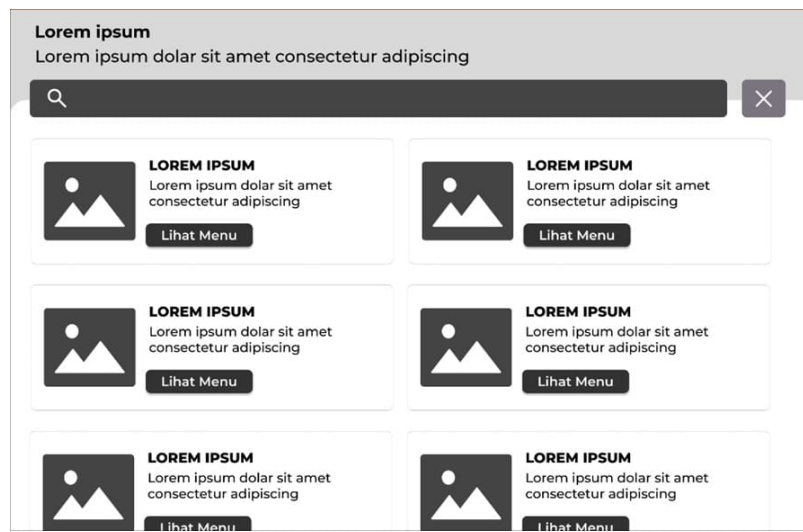
Kondisi Awal: Pembeli berada di halaman beranda



Gambar 4. 5 Wireframe Halaman Beranda

Aksi Aktor	Reaksi Sistem
1. Pada halaman beranda, pembeli melihat daftar menu terbaru (dari beberapa kios)	
2. Melihat beberapa kios yang tersedia	
3. Klik tombol lihat semua pada beranda	4. Menampilkan halaman berisi semua kios

Kondisi Akhir: Sistem menampilkan daftar kios di halaman kios-kios

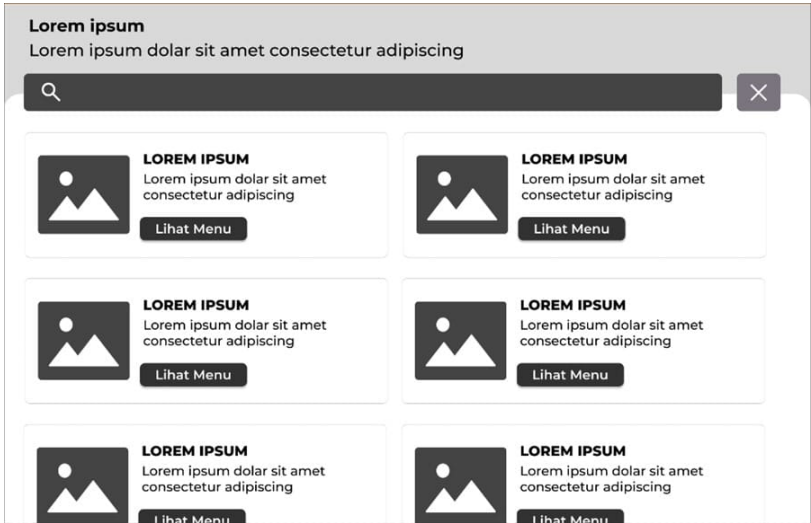
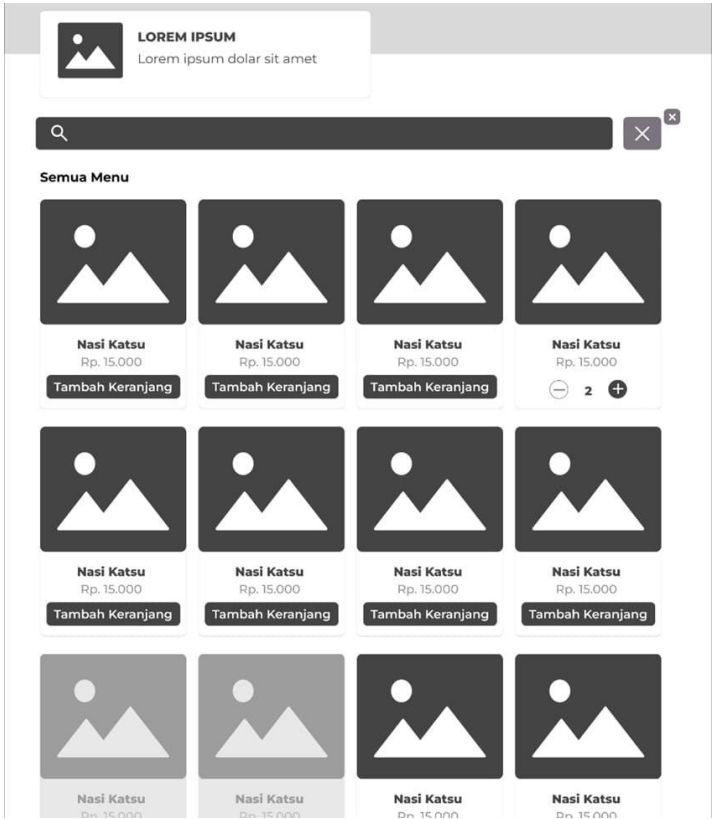


Gambar 4. 6 Wireframe Halaman Kios

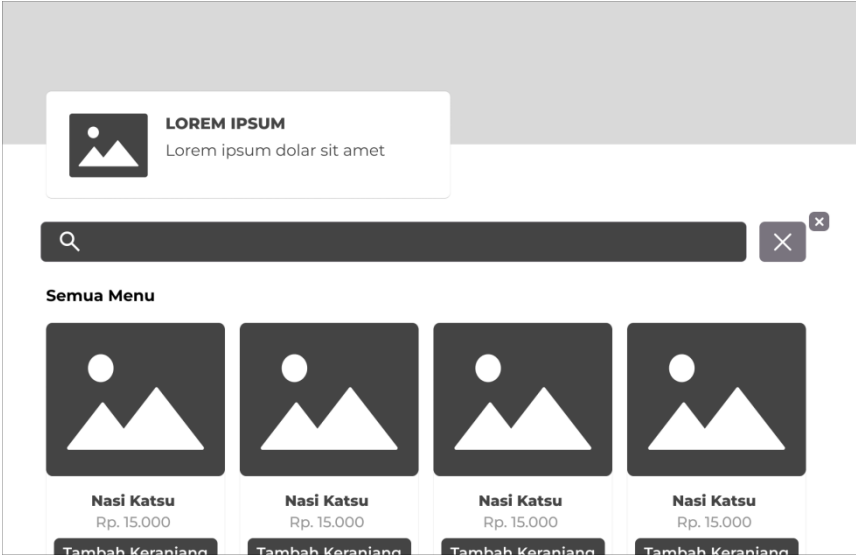
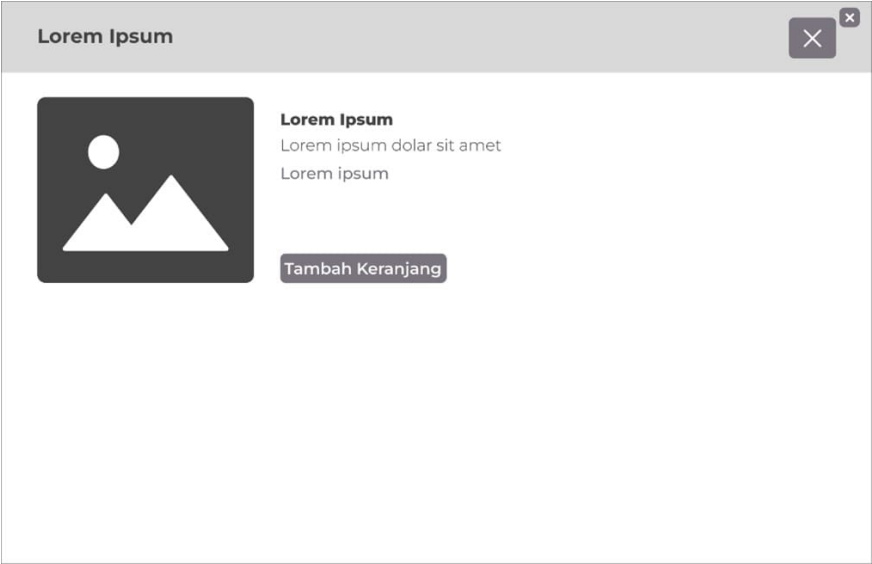
Skenario Alternatif Tidak Klik Tombol Lihat Semua	
1. Pada halaman beranda, pembeli tidak klik tombol lihat semua	
2. Memilih salah satu kios dari halaman beranda (yang tersedia)	

Tabel 4. 11 Skenario UC-02

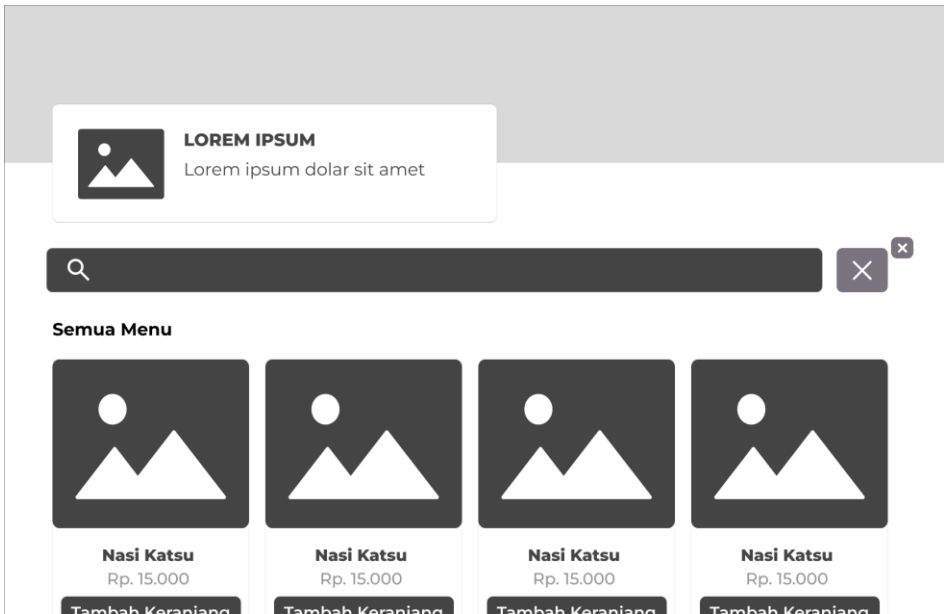
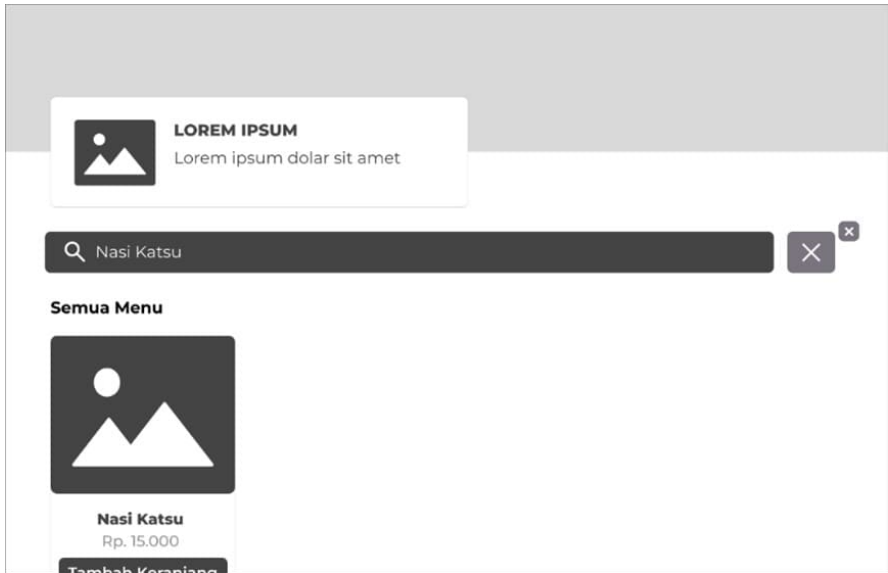
Identifikasi	
Kode	UC-02
Nama	Menampilkan daftar menu per kios
Tujuan	Pembeli dapat melihat daftar menu dari kios yang dipilih.
Deskripsi	Setelah memilih kios, pembeli dapat melihat daftar menu yang tersedia pada kios tersebut.
Aktor	Pembeli

Skenario Utama	
Kondisi Awal: Pembeli membuka halaman kios yang dipilih 	
Gambar 4. 7 Wireframe Halaman Kios	
Aksi Aktor	Reaksi Sistem
1. Mengklik "Lihat Menu" pada salah satu kios	2. Menampilkan daftar menu dari kios tersebut
3. Pembeli melihat daftar menu.	
Kondisi Akhir: Daftar menu kios tampil di halaman kios 	
Gambar 4. 8 Wireframe Halaman Menu	

Tabel 4. 12 Skenario UC-03

Identifikasi	
Kode	UC-03
Nama	Menampilkan detail menu
Tujuan	Pembeli dapat melihat informasi detail dari menu yang dipilih, seperti foto, nama menu, harga, dan deskripsi.
Deskripsi	Setelah pembeli memilih salah satu menu dari daftar menu di halaman kios, sistem menampilkan detail menu tersebut agar pembeli lebih jelas sebelum menentukan pilihan.
Aktor	Pembeli
Skenario Utama	
Kondisi Awal: Pembeli berada pada halaman kios, dan daftar menu kios sudah ditampilkan	
	
Gambar 4. 9 Wireframe Halaman Menu	
Aksi Aktor	Reaksi Sistem
1. Saat berada di halaman kios yang sudah dipilih, pembeli klik salah satu menu pada daftar menu	2. Menampilkan halaman detail menu berisi foto, nama menu, harga, dan deskripsi.
3. Pembeli melihat informasi detail menu.	
Kondisi Akhir: Detail menu tampil pada layar.	
	
Gambar 4. 10 Wireframe Halaman Detail Menu	

Tabel 4. 13 Skenario UC-04

Identifikasi	
Kode	UC-04
Nama	Mencari menu
Tujuan	Membantu pembeli menemukan menu tertentu dengan cepat sesuai kata kunci pencarian
Deskripsi	Saat di halaman kios, pembeli mengetik kata kunci di kolom pencarian. Sistem menampilkan menu sesuai kata kunci, atau pesan jika tidak ditemukan.
Aktor	Pembeli
Skenario Utama	
Kondisi Awal: Pembeli berada pada halaman kios yang berisi daftar menu	
	
Gambar 4. 11 Wireframe Halaman Menu	
Aksi Aktor	Reaksi Sistem
1. Pembeli mengetik kata kunci menu di kolom pencarian	2. Memproses kata kunci dan mencari menu yang sesuai.
	3. Sistem menampilkan daftar menu yang cocok dengan kata kunci.
4. Pembeli melihat hasil pencarian yang ditemukan	
Kondisi Akhir: Sistem menampilkan menu sesuai kata kunci.	
	
Gambar 4. 12 Wireframe Menampilkan Menu Berdasarkan Kata Kunci	

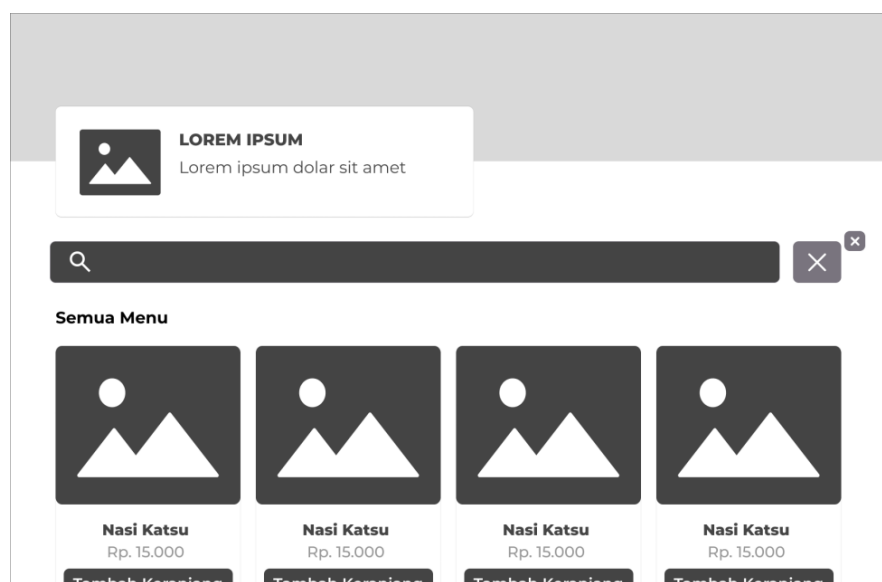
Skenario Alternatif – Menu tidak ditemukan	
1. Pembeli mengetik kata kunci yang tidak sesuai dengan menu manapun	2. Menampilkan pesan “Menu tidak ditemukan” dan tidak menampilkan daftar menu.

Tabel 4. 14 Skenario UC-05

Identifikasi	
Kode	UC-05
Nama	Mengelola keranjang
Tujuan	Pembeli menambahkan, memperbarui, atau menghapus menu di keranjang, sekaligus melihat isi dan total harga sebelum memesan.
Deskripsi	Dari halaman kios, pembeli dapat menambahkan menu ke keranjang, mengubah jumlah menu, atau menghapusnya. Pembeli juga bisa membuka halaman keranjang untuk melihat daftar menu beserta jumlah dan total harga.
Aktor	Pembeli

Skenario Utama

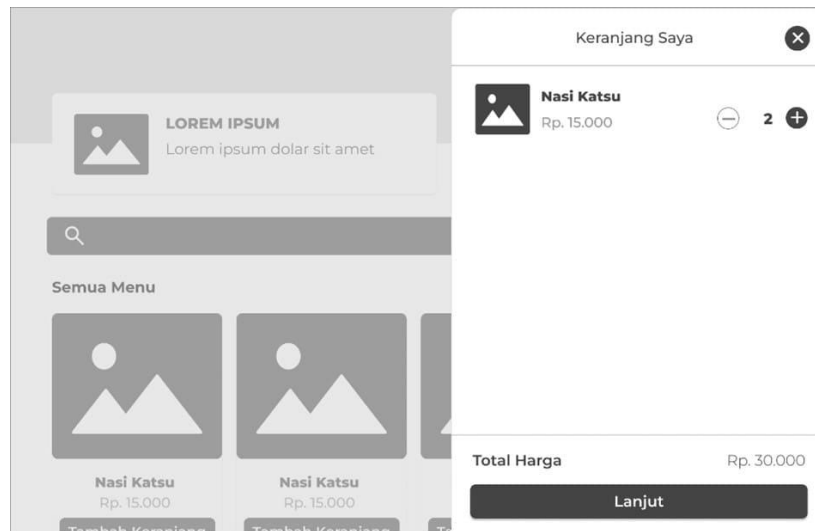
Kondisi Awal: Pembeli berada pada halaman kios yang berisi daftar menu



Gambar 4. 13 Wireframe Halaman Menu

Aksi Aktor	Reaksi Sistem
1. Pembeli menekan Tambah Keranjang pada card menu	2. Sistem menyimpan item ke keranjang.
	3. Sistem mengganti tombol pada card menjadi kontrol jumlah (+/-) dan menampilkan kuantitas.
4. Pembeli menambah/mengurangi jumlah via control	5. Sistem memperbarui keranjang (qty, subtotal, total harga).
6. Pembeli membuka halaman Keranjang	7. Sistem memuat & menampilkan daftar pesanan di keranjang (terbaru).

Kondisi Akhir: Menampilkan halaman keranjang yang sudah diperbarui sesuai pilihan pembeli (tambah/kurang)

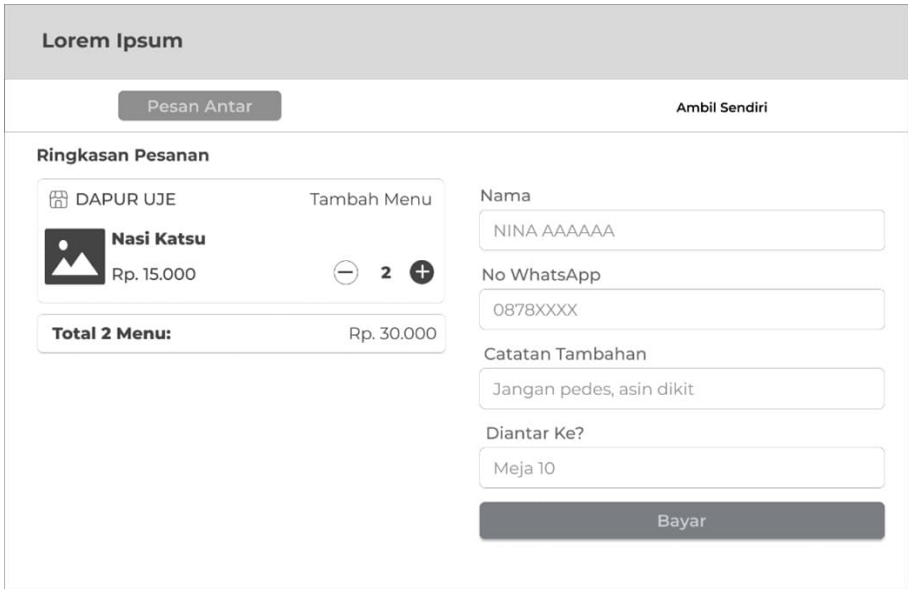


Gambar 4. 14 Wireframe Halaman Keranjang

Tabel 4. 15 Skenario UC-06

Identifikasi	
Kode	UC-06
Nama	Membuat pesanan dari keranjang
Tujuan	Membantu pembeli membuat pesanan berdasarkan isi keranjang dengan melengkapi data diri dan memilih tipe pengantaran.
Deskripsi	Setelah memeriksa keranjang, pembeli melanjutkan ke konfirmasi pesanan dengan mengisi data dan memilih tipe pengantaran. Jika diantar, pembeli mengisi tempat pengantaran
Aktor	Pembeli
Skenario Utama	
Kondisi Awal: Pembeli berada di halaman keranjang dan isi keranjang sudah sesuai.	
Aksi Aktor	Reaksi Sistem
1. Pembeli klik tombol Lanjut dari halaman keranjang	2. Menampilkan Halaman detail pesanan (ringkasan pesanan, input data pemesanan & pilihan jenis pengantaran)
3. Melihat ringkasan pesanan (jumlah menu dan total harga)	
4. Memilih jenis pengantaran (pesan antar)	

Gambar 4. 15 Wireframe Halaman Keranjang

5. Mengisi input data pemesanan berupa Nama, No WhatsApp, Catatan, Diantarkan Kemana?	6. Menyimpan data pemesanan
	7. Menampilkan tombol bayar aktif
Kondisi Akhir: Menampilkan halaman konfirmasi detail pesanan dengan tombol bayar aktif	
	
Gambar 4. 16 Wireframe Halaman Konfirmasi Detail Pesanan	
Skenario Alternatif – Ambil Sendiri	
1. Pada halaman detail pesanan, pembeli mengisi data pemesanan berupa Nama, No WhatsApp, Catatan	2. Menyimpan data pemesanan
	3. Menampilkan tombol bayar aktif

Tabel 4. 16 Skenario UC-07

Identifikasi	
Kode	UC-07
Nama	Menyediakan <i>Payment Gateway</i>
Tujuan	Memfasilitasi pembeli dalam melakukan pembayaran melalui <i>payment gateway</i>
Deskripsi	Pembeli dapat melakukan pembayaran dengan QRIS, transfer bank, atau e-wallet melalui <i>payment gateway</i> (Midtrans). Setelah pembayaran berhasil diverifikasi, sistem akan menampilkan pop up konfirmasi berhasil.
Aktor	Pembeli
Skenario Utama	

Kondisi Awal: Pembeli melihat tombol bayar aktif

Gambar 4. 17 Wireframe Halaman Konfirmasi Detail Pesanan Ketika Tombol Bayar Aktif

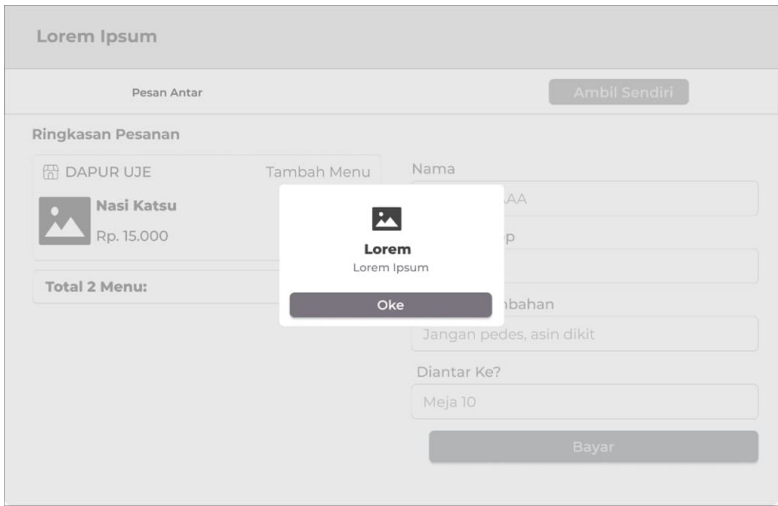
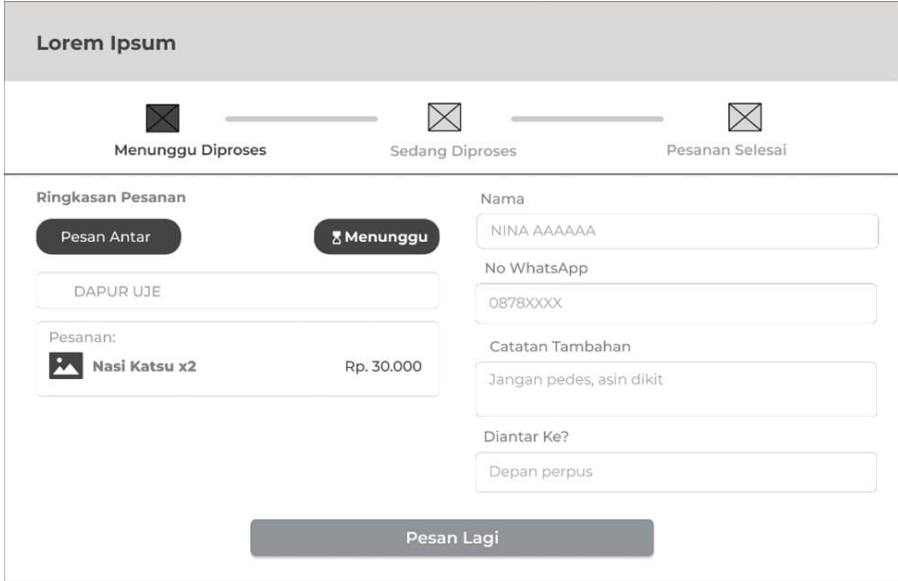
Aksi Aktor	Reaksi Sistem
1. Klik tombol bayar	2. Menampilkan halaman pembayaran (<i>payment gateway</i> , contoh: Midtrans).
3. Memilih metode pembayaran (QRIS, transfer bank, atau e-wallet).	
4. Melakukan pembayaran sesuai metode yang dipilih.	5. Memverifikasi status pembayaran.
6. Memeriksa status pembayaran	7. Menampilkan pop up "Pembayaran Berhasil".
	8. Menampilkan tombol "Oke" pada pop up.
9. Klik tombol "Oke" pada pop up.	

Kondisi Akhir: Menampilkan pop up pembayaran berhasil dan klik tombol oke pada popup

Gambar 4. 18 Wireframe Pop Up Pembayaran Berhasil

Tabel 4. 17 Skenario UC-08

Identifikasi	
Kode	UC-08
Nama	Menampilkan status dan estimasi waktu pesanan
Tujuan	Memberikan informasi status pesanan dan estimasi waktu selesai kepada pembeli
Deskripsi	Pembeli dapat memantau status pesanan yang sedang diproses beserta estimasi waktu penyelesaian
Aktor	Pembeli

Skenario Utama	
Kondisi Awal: Pembeli klik tombol ok pada pop up "Pembayaran Berhasil"	
	
Gambar 4. 19 Wireframe Pop Up Pembayaran Berhasil	
Aksi Aktor	Reaksi Sistem
1. Melihat halaman status pesanan	2. Menampilkan status pesanan pertama kali "Menunggu diproses"
	3. Menampilkan estimasi waktu (countdown)
4. Memantau estimasi waktu pesanan	
5. Memantau perubahan status	6. Memperbarui status pesanan secara otomatis (Menunggu, Sedang diproses dan Selesai)
Kondisi Akhir: Sistem menampilkan halaman status pesanan dengan estimasi waktu dan status terbaru	
	
Gambar 4. 20 Wireframe Halaman Status Pesanan	

Tabel 4. 18 Skenario UC-09

Identifikasi	
Kode	UC-09
Nama	Menerima notifikasi pesanan selesai
Tujuan	Memberikan informasi kepada pembeli bahwa pesanan sudah selesai.
Deskripsi	Menerima notifikasi pesanan selesai kepada pembeli melalui WhatsApp, lalu memperbarui status pesanan pada halaman status pesanan dan tombol aktif Pesan Lagi
Aktor	Pembeli

Skenario Utama	
Kondisi Awal: Menampilkan halaman status pesanan sedang dalam proses dan belum selesai.	

Gambar 4. 21 Wireframe Halaman Status Pesanan Dengan Status Sedang Diproses

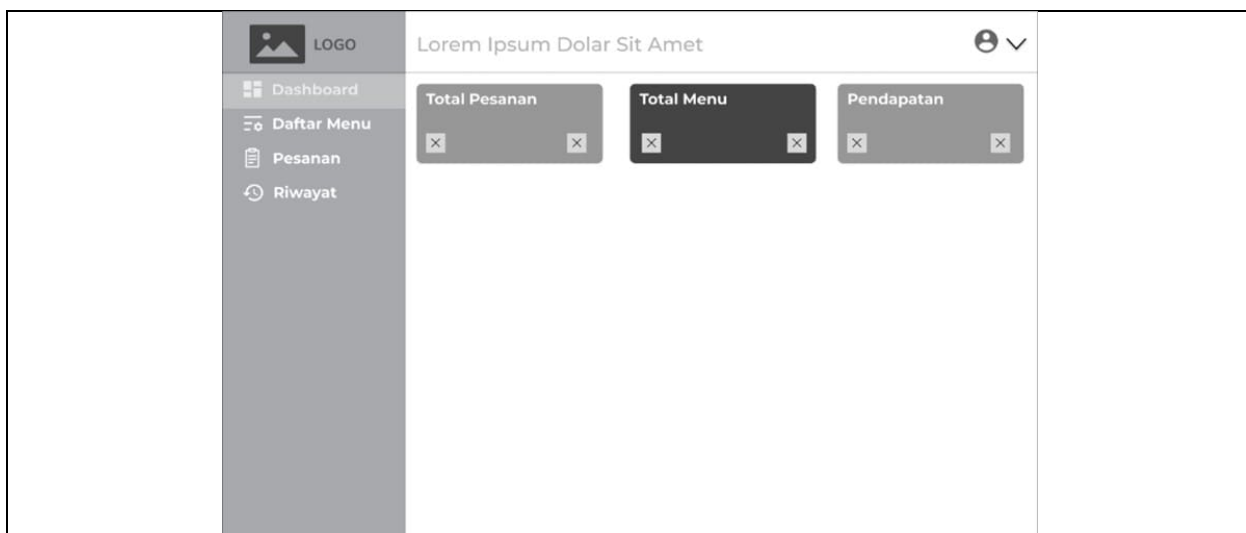
Aksi Aktor	Reaksi Sistem
1. Memantau halaman status pesanan	2. Memperbarui status pesanan nya menjadi pesanan selesai
4. Menerima notifikasi WhatsApp	3. Menampilkan tombol Pesan Lagi aktif

Kondisi Akhir: Sistem menampilkan halaman status pesanan dengan tombol aktif Pesan Lagi

Gambar 4. 22 Wireframe Halaman Status Pesanan Dengan Status Selesai Dan Tombol Aktif

Tabel 4. 19 Skenario UC-10

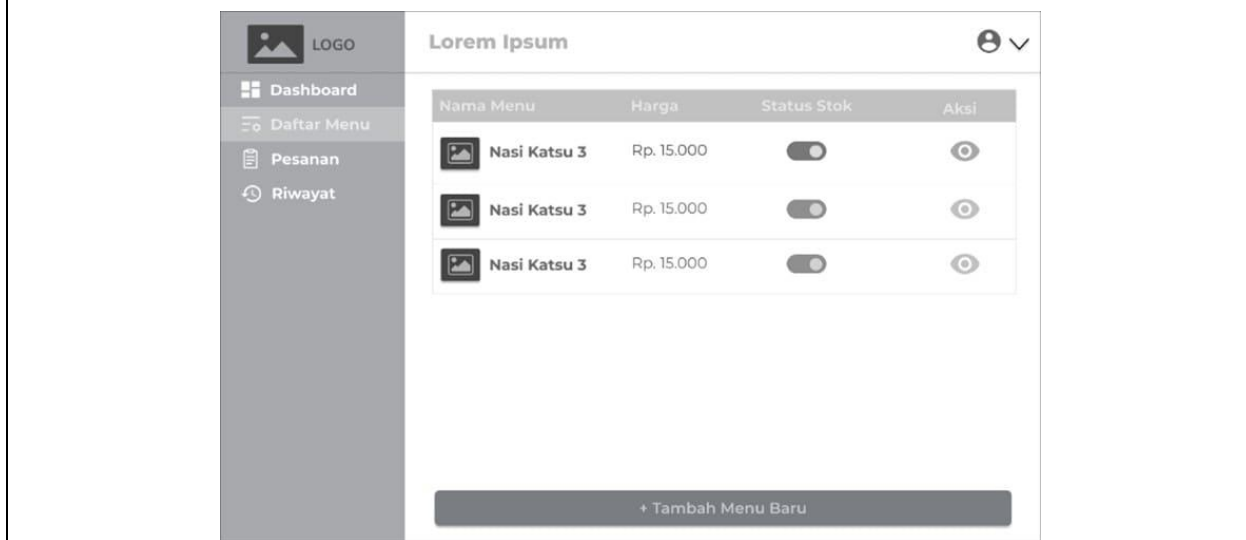
Identifikasi	
Kode	UC-10
Nama	Mengelola Menu
Tujuan	Memastikan informasi menu yang ditampilkan ke pembeli selalu akurat dan terbaru.
Deskripsi	Penjual dapat menambahkan, mengubah, atau menghapus menu yang tersedia di sistem.
Aktor	Penjual
Skenario Utama	
Kondisi Awal: Penjual membuka halaman dashboard	



Gambar 4. 23 Wireframe Halaman Dashboard

Aksi Aktor	Reaksi Sistem
1. Pada halaman dashboard, penjual klik menu “daftar menu”	2. Menampilkan halaman daftar menu
3. Klik tombol “tambah menu baru”	4. Menampilkan form tambah menu baru
5. Mengisi form menu baru, berupa gambar, nama, deskripsi, harga dan waktu penyajian	
6. Klik tombol “simpan” untuk menyimpan form data menu baru yang telah diisi	7. Menyimpan data menu baru
8. Melakukan edit dan hapus data menu	9. Menyimpan perubahan dan menampilkan daftar menu terbaru.

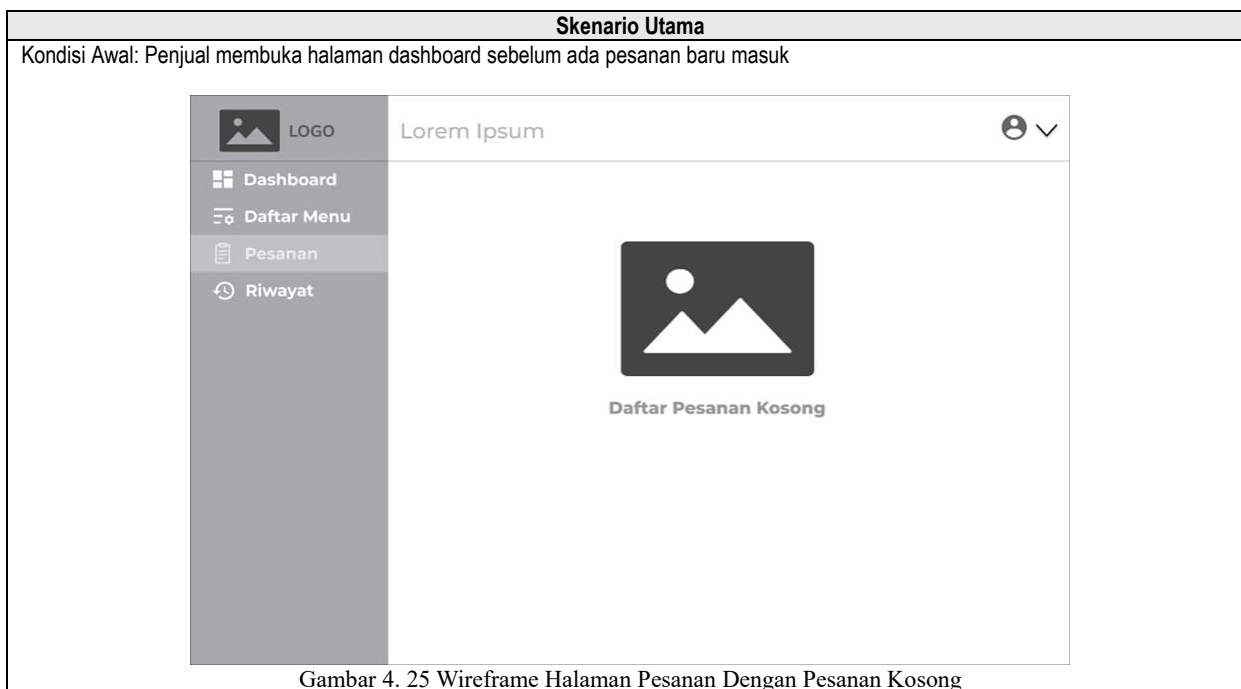
Kondisi Akhir: Sistem menampilkan halaman daftar menu yang sudah diperbarui oleh penjual



Gambar 4. 24 Wireframe Daftar Menu Dashboard

Tabel 4. 20 Skenario UC-11

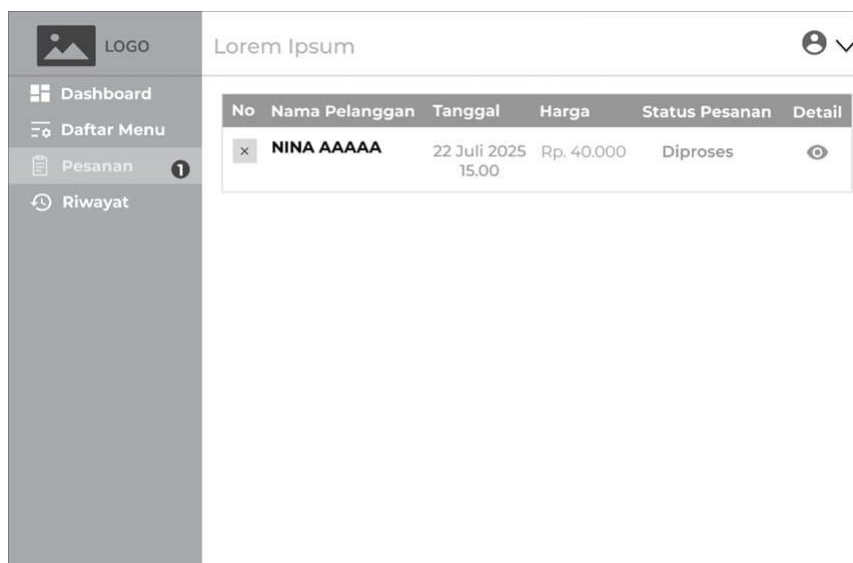
Identifikasi	
Kode	UC-11
Nama	Menerima Notifikasi Pesanan Masuk
Tujuan	Memberi tahu pembeli bahwa ada pesanan baru yang masuk agar dapat segera dilihat.
Deskripsi	Menampilkan notifikasi pesanan baru kepada pembeli dalam bentuk link. Saat link diklik, pembeli diarahkan ke dashboard yang menampilkan daftar pesanan masuk.
Aktor	Penjual



Gambar 4. 25 Wireframe Halaman Pesanan Dengan Pesanan Kosong

Aksi Aktor	Reaksi Sistem
1. Penjual menerima notifikasi WhatsApp berisi link pesanan baru masuk	
2. Klik link	
3. Melakukan login kembali	4. Menampilkan halaman dashboard dengan badge jumlah pesanan baru

Kondisi Akhir: menampilkan daftar pesanan baru di dashboard



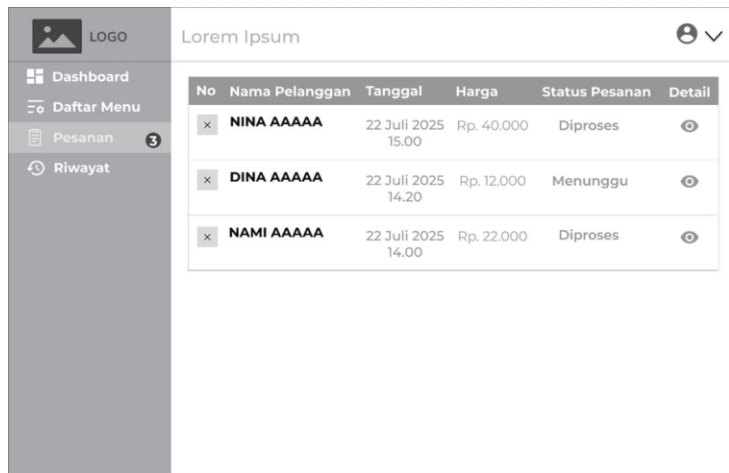
Gambar 4. 26 Wireframe Halaman Pesanan Dengan Daftar Pesanan Masuk

Tabel 4. 21 Skenario UC-12

Identifikasi	
Kode	UC-12
Nama	Menampilkan Daftar Pesanan dan Detail Pesanan
Tujuan	Membantu penjual melihat dan memproses pesanan dengan lengkap.
Deskripsi	Menampilkan daftar pesanan yang masuk beserta detail seperti menu, jumlah, catatan pembeli, dan total harga.
Aktor	Penjual

Skenario Utama

Kondisi Awal: Penjual membuka halaman daftar pesanan

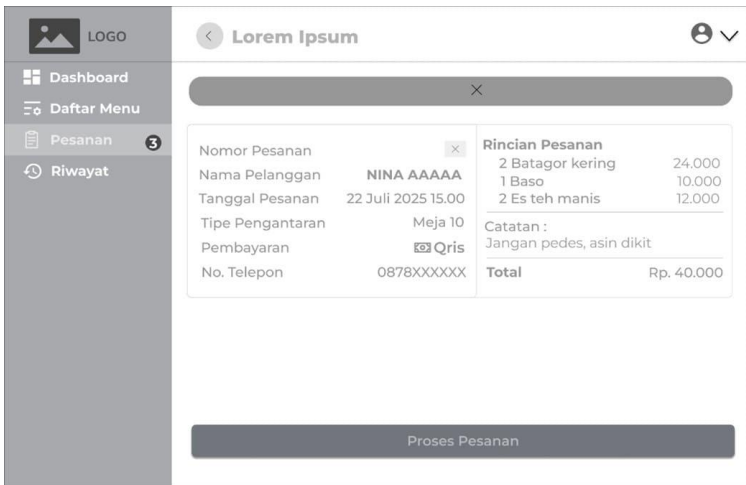


The wireframe shows a web application layout for a seller's order management page. On the left is a dark sidebar with navigation links: 'Dashboard' (grid icon), 'Daftar Menu' (list icon), 'Pesanan' (document icon with a red badge showing '3'), and 'Riwayat' (clock icon). The main content area has a header with a 'LOGO' placeholder, the text 'Lorem Ipsum', and a user profile icon with a dropdown arrow. Below the header is a table with the following data:

No	Nama Pelanggan	Tanggal	Harga	Status Pesanan	Detail
x	NINA AAAAA	22 Juli 2025 15.00	Rp. 40.000	Diproses	👁
x	DINA AAAAA	22 Juli 2025 14.20	Rp. 12.000	Menunggu	👁
x	NAMI AAAAA	22 Juli 2025 14.00	Rp. 22.000	Diproses	👁

Gambar 4. 27 Wireframe Halaman Pesanan

Gambar 4. 27 Wireframe Halaman Pesanan

Aksi Aktor	Reaksi Sistem																		
Penjual melihat daftar semua pesanan yang masuk																			
Memilih salah satu pesanan sesuai urutan untuk melihat detail dan memproses pesanan	Menampilkan detail daftar pesanan yang masuk beserta detail seperti menu, jumlah, catatan pembeli, dan total harga.																		
Kondisi Akhir: menampilkan detail pesanan dengan lengkap																			
 <table border="1"> <thead> <tr> <th colspan="2">Rincian Pesanan</th></tr> </thead> <tbody> <tr> <td>Nomor Pesanan</td><td>x</td></tr> <tr> <td>Nama Pelanggan</td><td>NINA AAAAA</td></tr> <tr> <td>Tanggal Pesanan</td><td>22 Juli 2025 15.00</td></tr> <tr> <td>Tipe Pengantaran</td><td>Meja 10</td></tr> <tr> <td>Pembayaran</td><td>Qris</td></tr> <tr> <td>No. Telepon</td><td>0878XXXXXX</td></tr> <tr> <td colspan="2">Catatan : Jangan pedes, asin dikit</td></tr> <tr> <td>Total</td><td>Rp. 40.000</td></tr> </tbody> </table> Proses Pesanan		Rincian Pesanan		Nomor Pesanan	x	Nama Pelanggan	NINA AAAAA	Tanggal Pesanan	22 Juli 2025 15.00	Tipe Pengantaran	Meja 10	Pembayaran	Qris	No. Telepon	0878XXXXXX	Catatan : Jangan pedes, asin dikit		Total	Rp. 40.000
Rincian Pesanan																			
Nomor Pesanan	x																		
Nama Pelanggan	NINA AAAAA																		
Tanggal Pesanan	22 Juli 2025 15.00																		
Tipe Pengantaran	Meja 10																		
Pembayaran	Qris																		
No. Telepon	0878XXXXXX																		
Catatan : Jangan pedes, asin dikit																			
Total	Rp. 40.000																		

Gambar 4. 28 Wiraframe Halaman Detail Pesanan

Tabel 4. 22 Skenario UC-13

Identifikasi	
Kode	UC-13
Nama	Mengubah Status Pesanan
Tujuan	Memberikan informasi status pesanan yang akurat kepada pembeli
Deskripsi	Penjual memperbarui status pesanan (menunggu diproses, sedang diproses, selesai) agar pembeli mengetahui update status pesanannya
Aktor	Penjual
Skenario Utama	

Kondisi Awal: Penjual membuka halaman detail pesanan yang akan di proses

Wireframe of the initial order detail page. The page has a sidebar with 'Pesanan' (3) and 'Riwayat'. The main content shows order details for 'NINA AAAAA' on '22 Juli 2025 15.00'. The order items are 2 Batagor kering (24,000), 1 Baso (10,000), and 2 Es teh manis (12,000). The total is Rp. 40.000. A 'Proses Pesanan' button is at the bottom.

Gambar 4. 29 Wiraframe Halaman Detail Pesanan Dengan Status Menunggu Diproses

Aksi Aktor	Reaksi Sistem
1. Klik tombol proses pesanan yang ada di halaman detail pesanan	2. Mengubah status pesanan menjadi "pesanan di proses"
3. Memasak atau menyiapkan pesanan	.

Kondisi Akhir: Sistem menampilkan halaman pesanan detail pesanan dengan status pesanan sedang di proses

Wireframe of the order detail page after processing. The 'Proses Pesanan' button has been replaced with 'Kirim Pesanan'. The rest of the page content remains the same.

Gambar 4. 30 Wiraframe Halaman Detail Pesanan Dengan Status Pesanan Diproses

Tabel 4. 23 Skenario UC-14

Identifikasi	
Kode	UC-14
Nama	Menampilkan Tipe Pengantaran
Tujuan	Penjual mengetahui tipe pengantaran pesanan (diantar atau ambil sendiri) agar bisa menyerahkan pesanan sesuai pilihan pembeli.
Deskripsi	Perangkat lunak menampilkan tampilan opsi metode pengantaran (Ambil Sendiri atau Diantar). Sistem menyesuaikan tombol aksi yang tersedia berdasarkan tipe pengantaran.
Aktor	Pembeli

Skenario Utama	
Kondisi Awal: Penjual membuka halaman detail pesanan yang sedang diproses	

Gambar 4. 31 Wireframe Halaman Detail Pesanan Dengan Status Pesanan Diproses

Aksi Aktor	Reaksi Sistem
1. Penjual mengecek tipe pengantaran pesanan	2. Menampilkan tipe pengantaran yang dipilih pembeli
3. Penjual menekan tombol aksi Kirim Pesanan (jika tipe Diantar)	4. Menandai proses pengantaran dimulai
5. Penjual menyerahkan pesanan	6. Menampilkan tombol aktif Pesanan Sudah Diantar
7. Klik tombol Pesanan Sudah Diantar	8. Mengubah status pesanan menjadi "Pesanan Selesai"

Kondisi Akhir: Sistem menampilkan daftar pesanan selesai di halaman riwayat pesanan

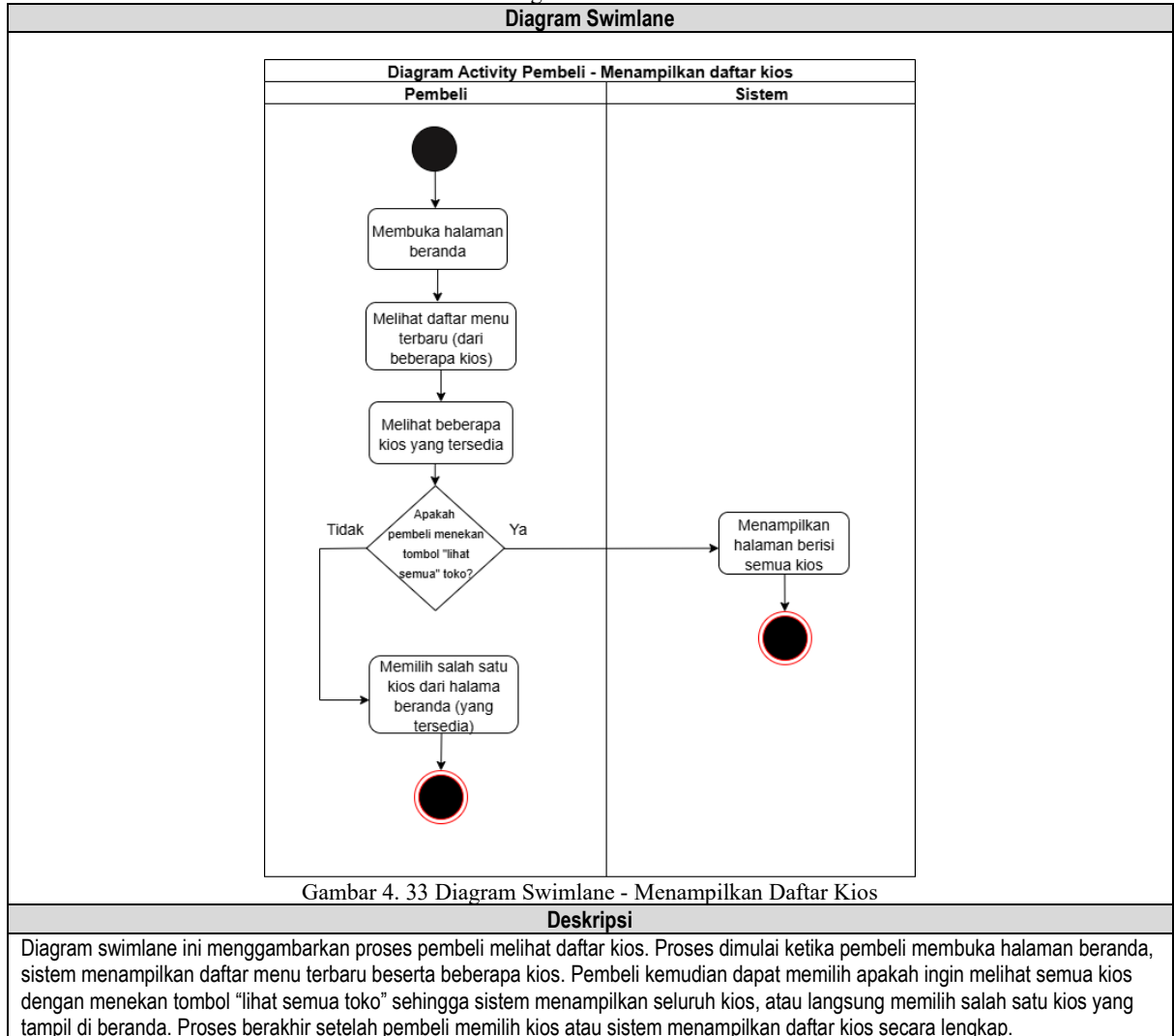
Gambar 4. 32 Wireframe Halaman Riwayat Pesanan

Skenario Alternatif – Ambil Sendiri	
1. Penjual menekan tombol aksi Pesanan Siap Diambil (jika tipe Diambil)	2. Menandai proses pengambilan dimulai
3. Penjual menyerahkan pesanan	4. Menampilkan tombol aktif Pesanan Sudah Diambil
5. Klik tombol Pesanan Sudah Diambil	6. Mengubah status pesanan menjadi "Pesanan Selesai"

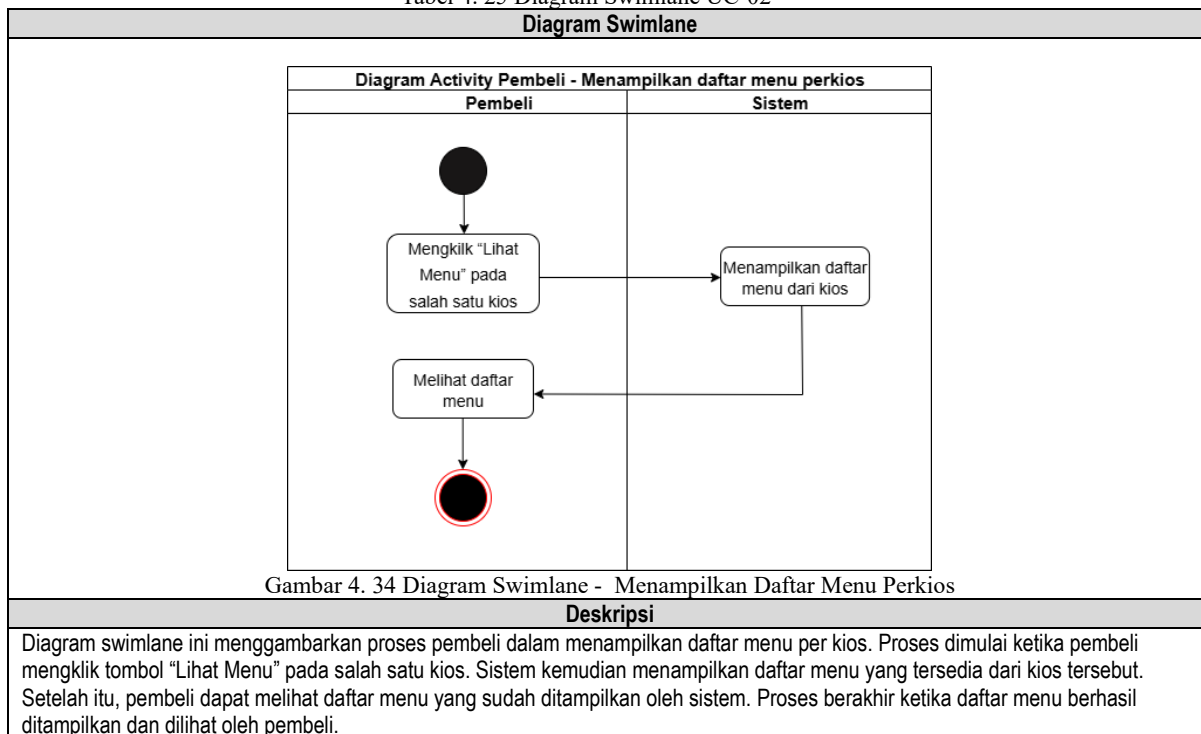
4.2.5 Diagram Swimlane

Diagram swimlane merupakan jenis diagram yang digunakan untuk memperlihatkan alur suatu proses, dengan menekankan peran serta interaksi antar bagian yang terlibat. Diagram ini juga menunjukkan bagaimana suatu proses berpindah melalui beberapa tahap yang berbeda.

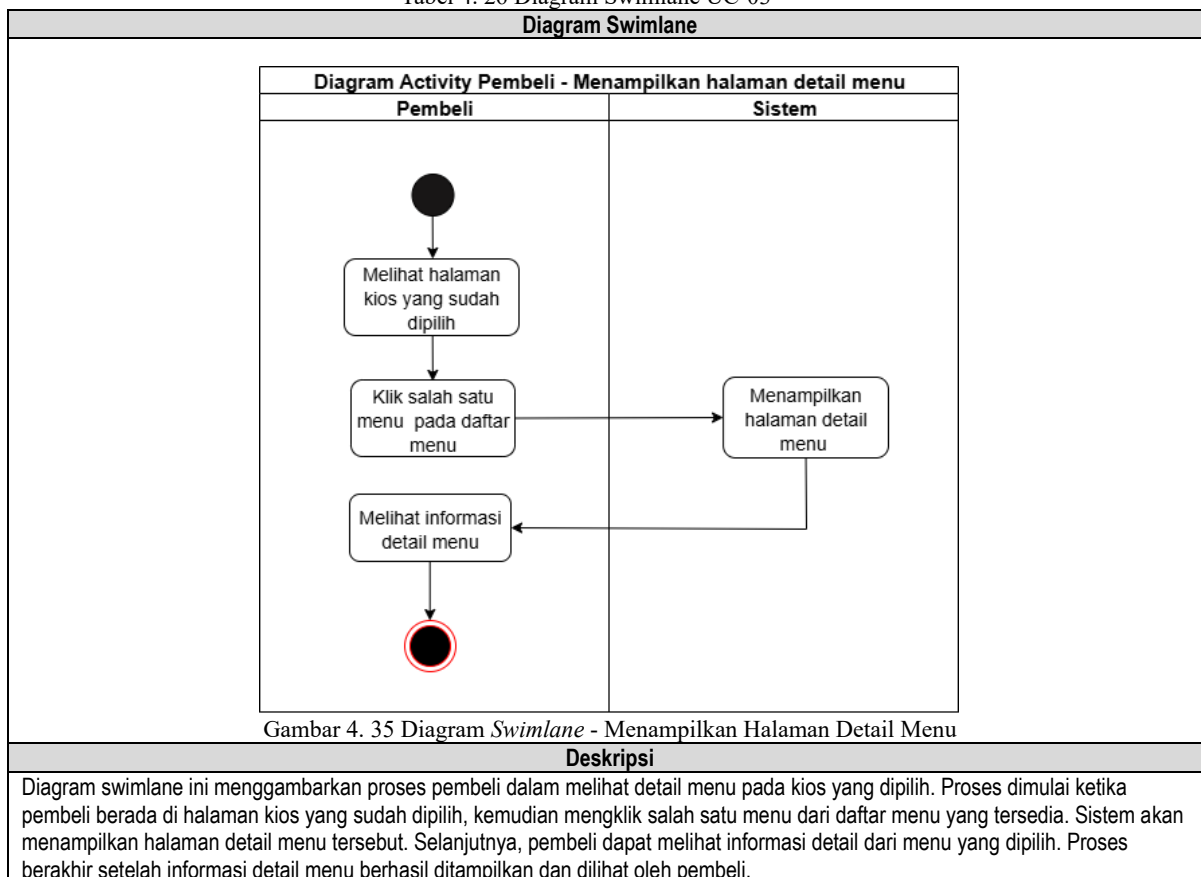
Tabel 4. 24 Diagram Swimlane UC-01



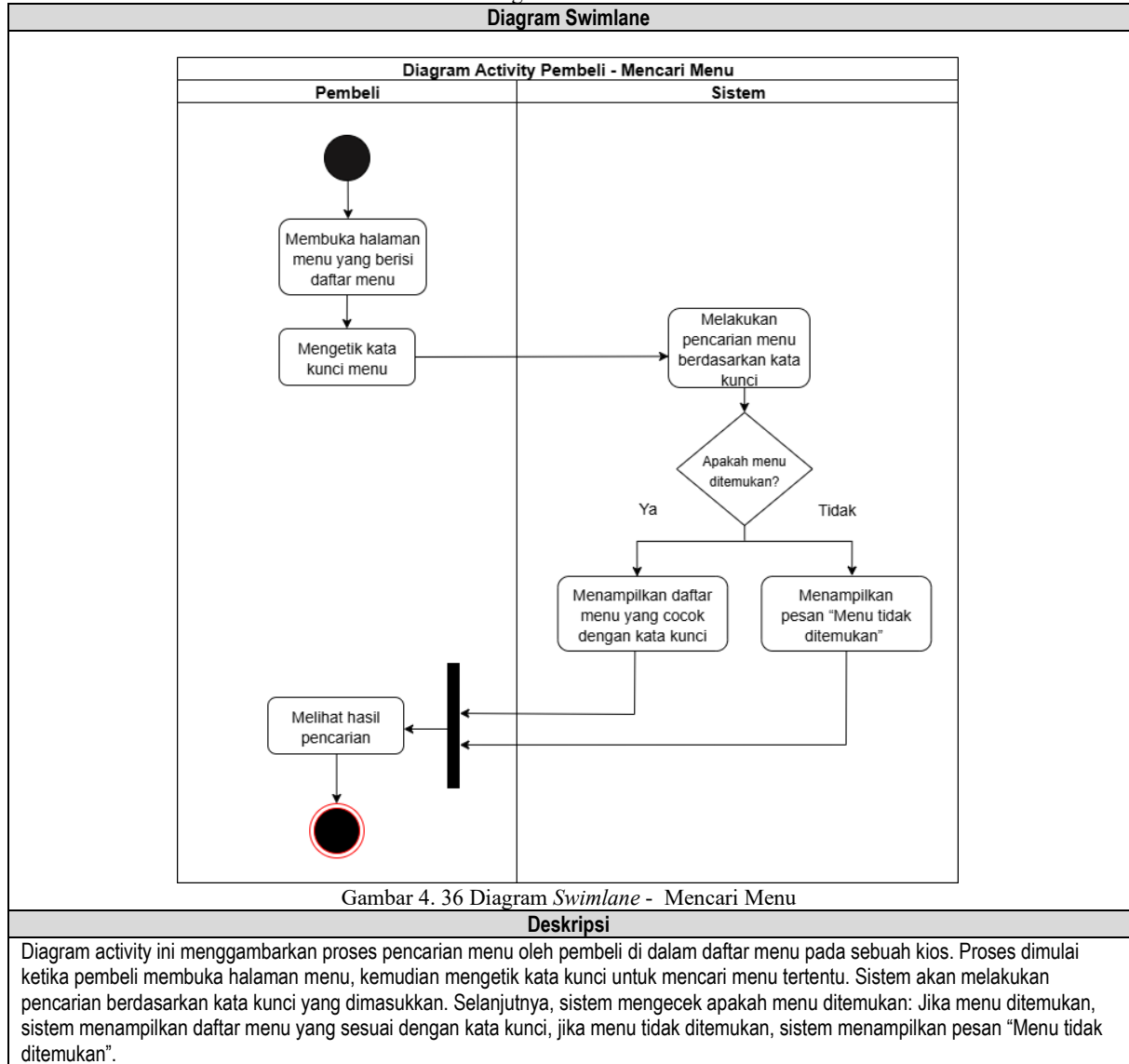
Tabel 4. 25 Diagram Swimlane UC-02



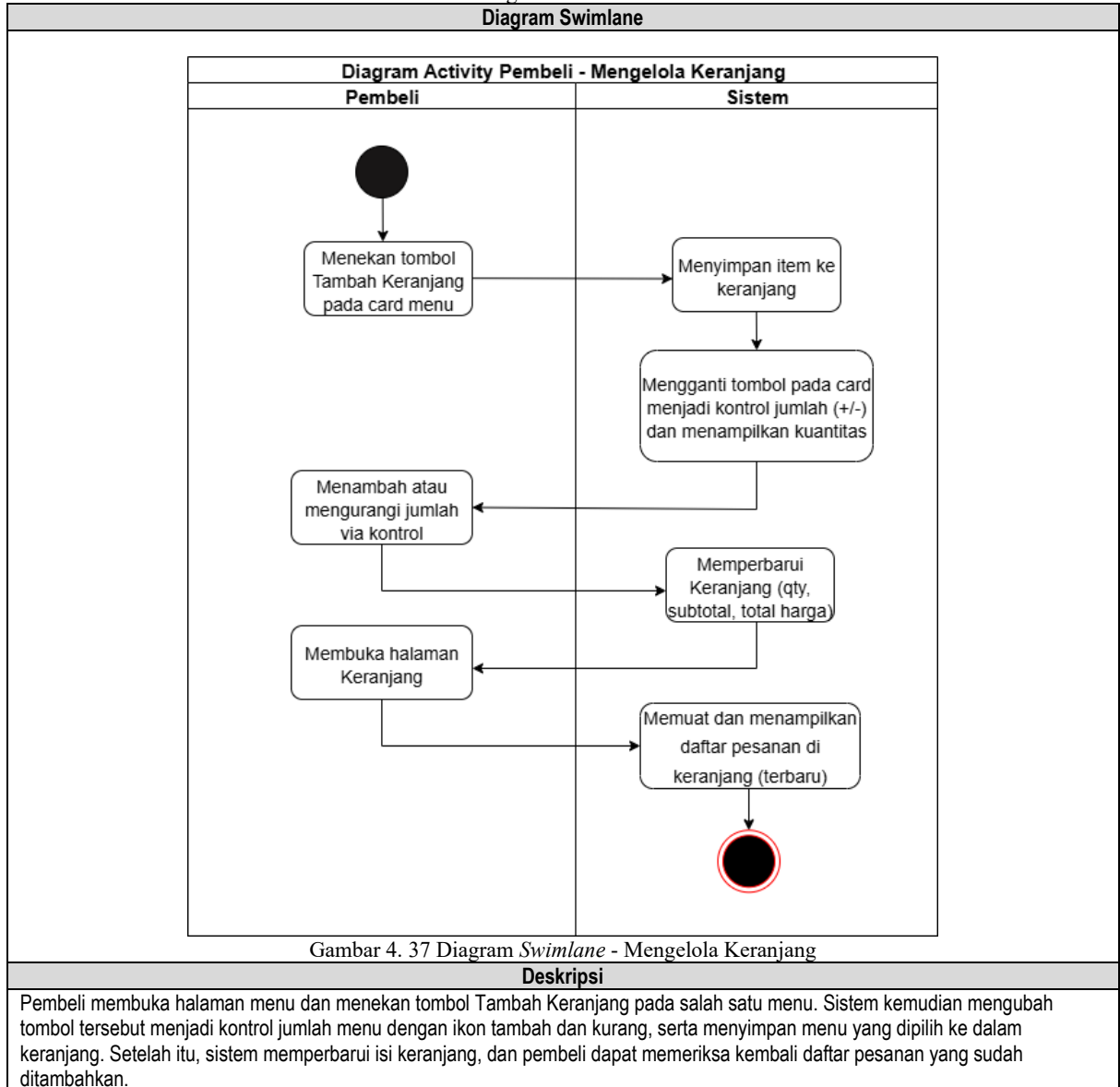
Tabel 4. 26 Diagram Swimlane UC-03



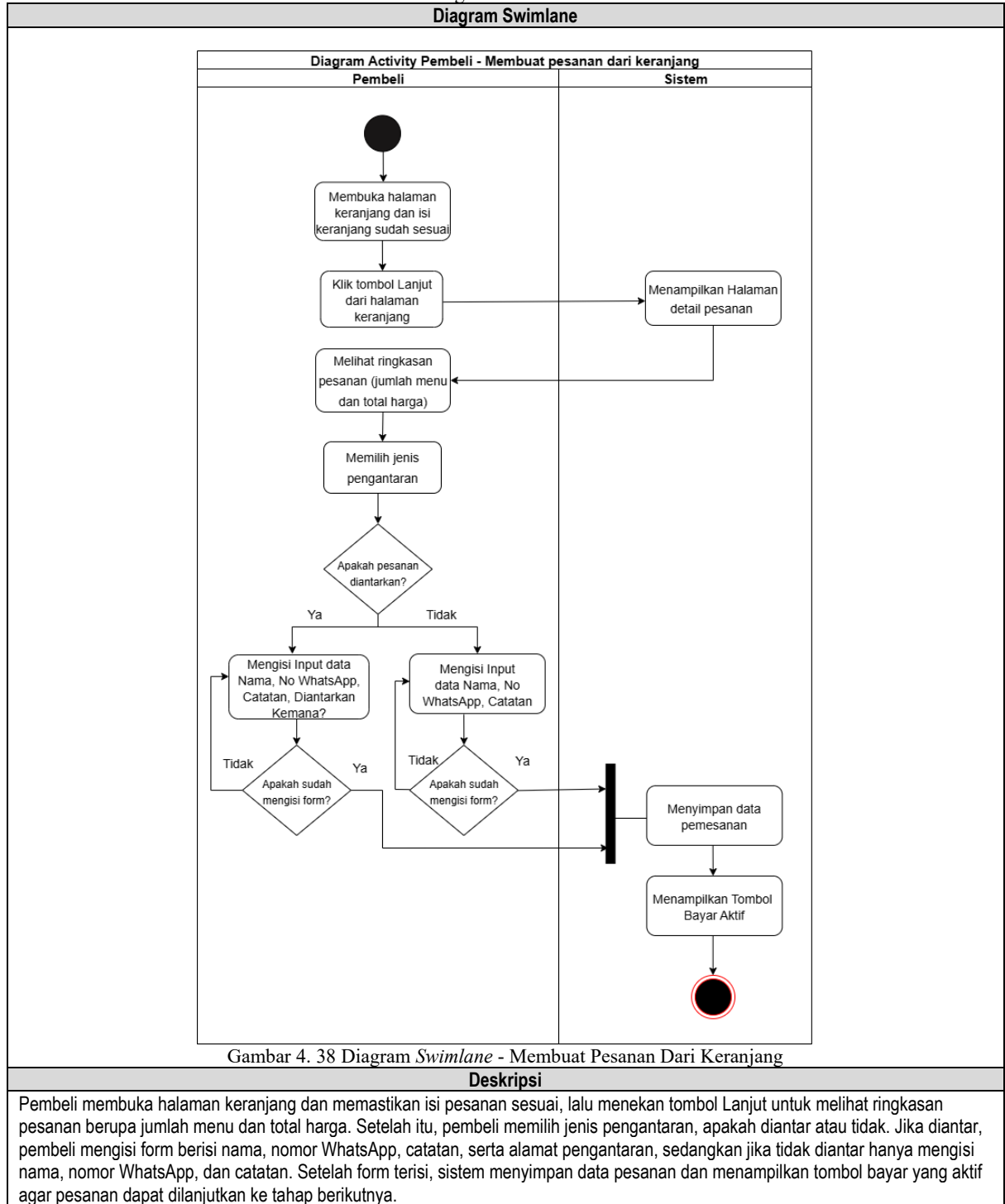
Tabel 4. 27 Diagram Swimlane UC-04



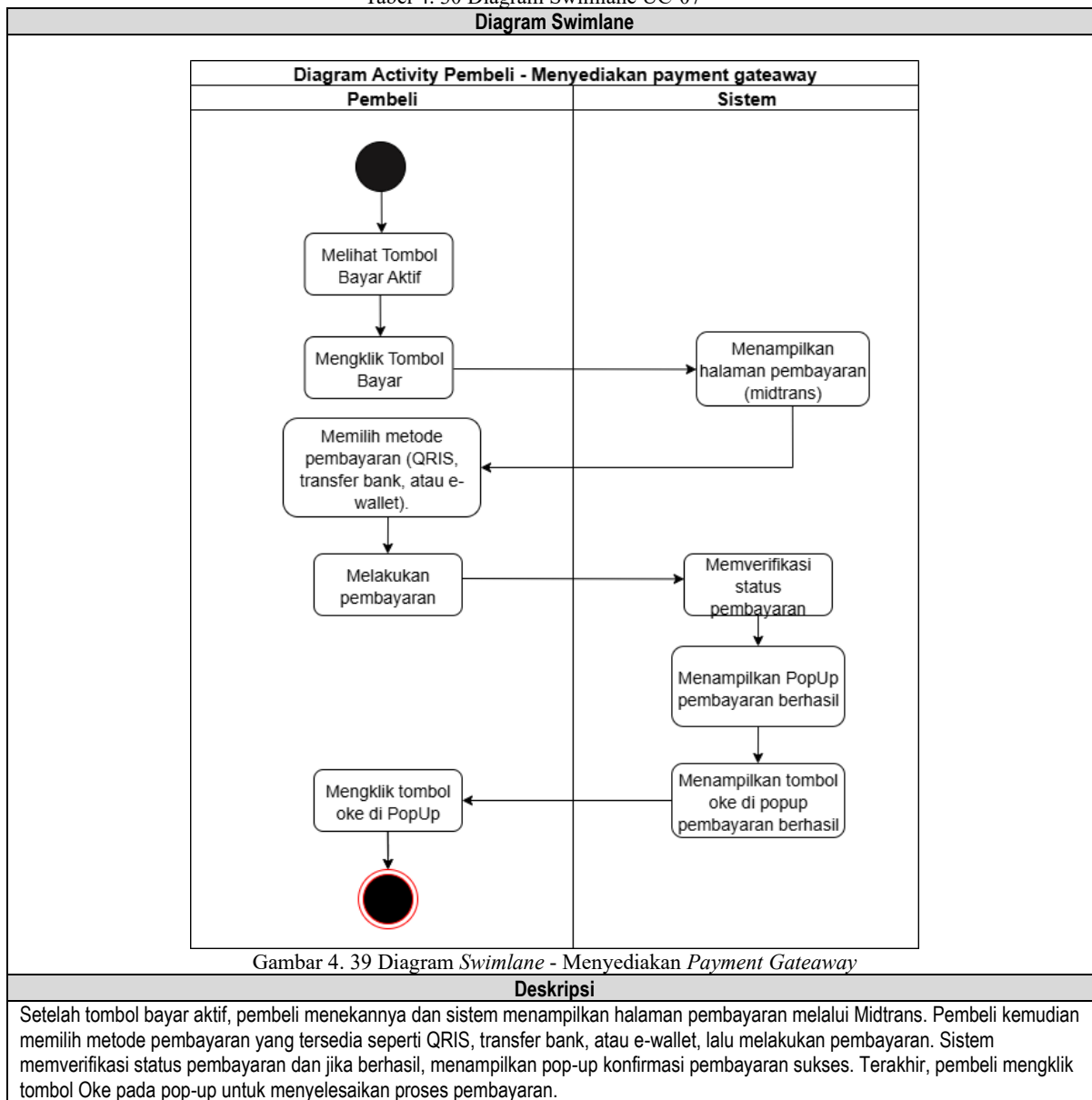
Tabel 4. 28 Diagram Swimlane UC-05



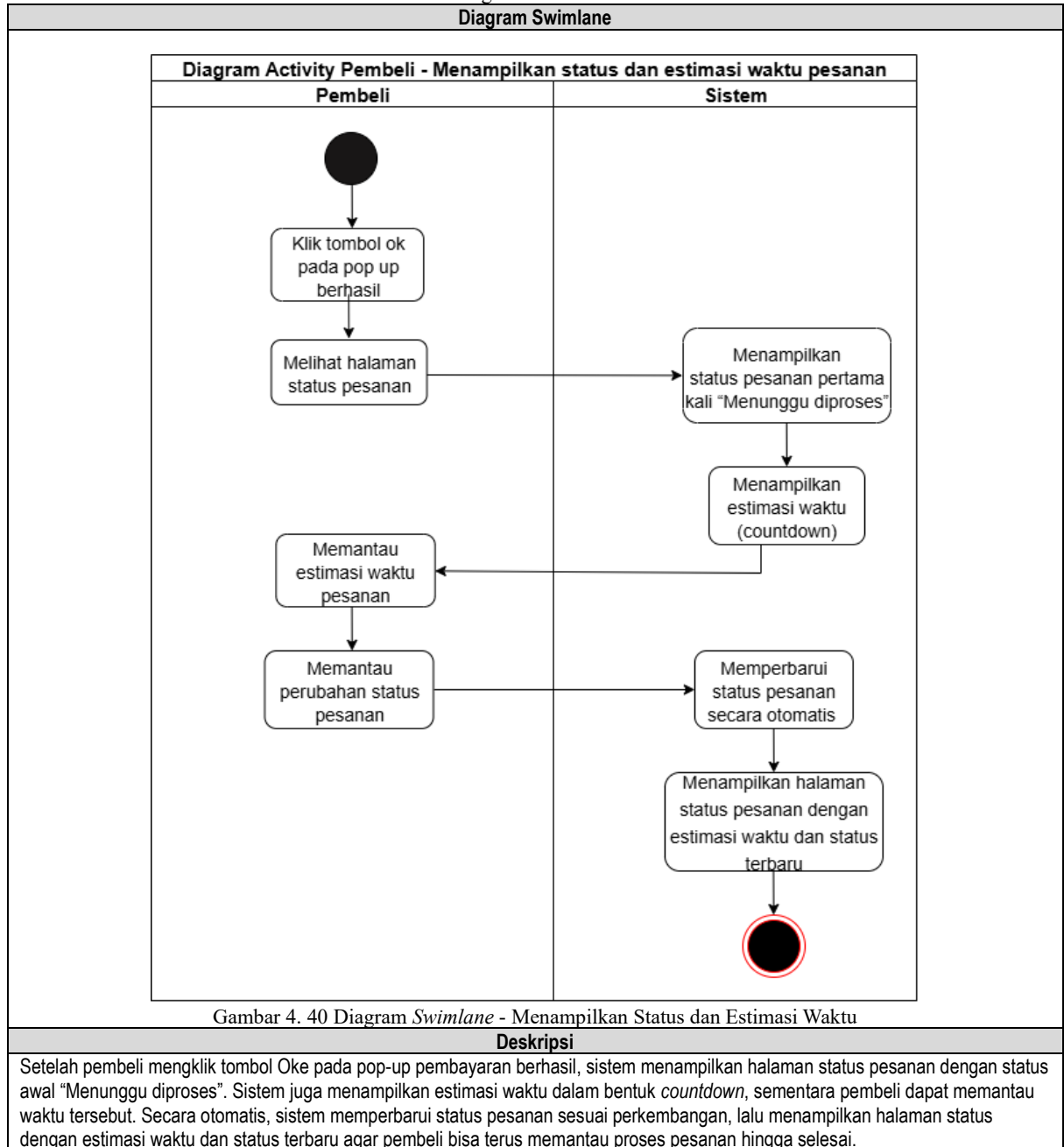
Tabel 4. 29 Diagram Swimlane UC-06



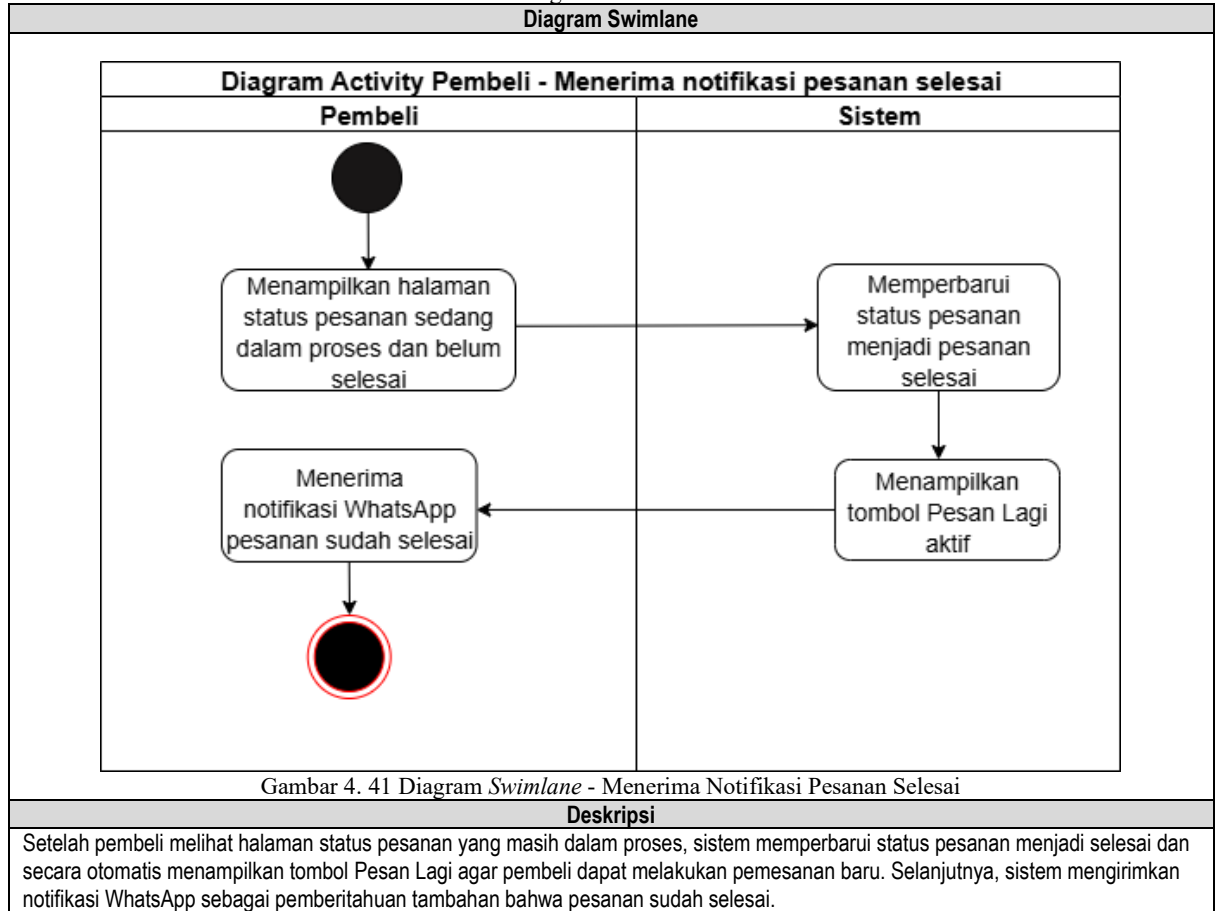
Tabel 4. 30 Diagram Swimlane UC-07



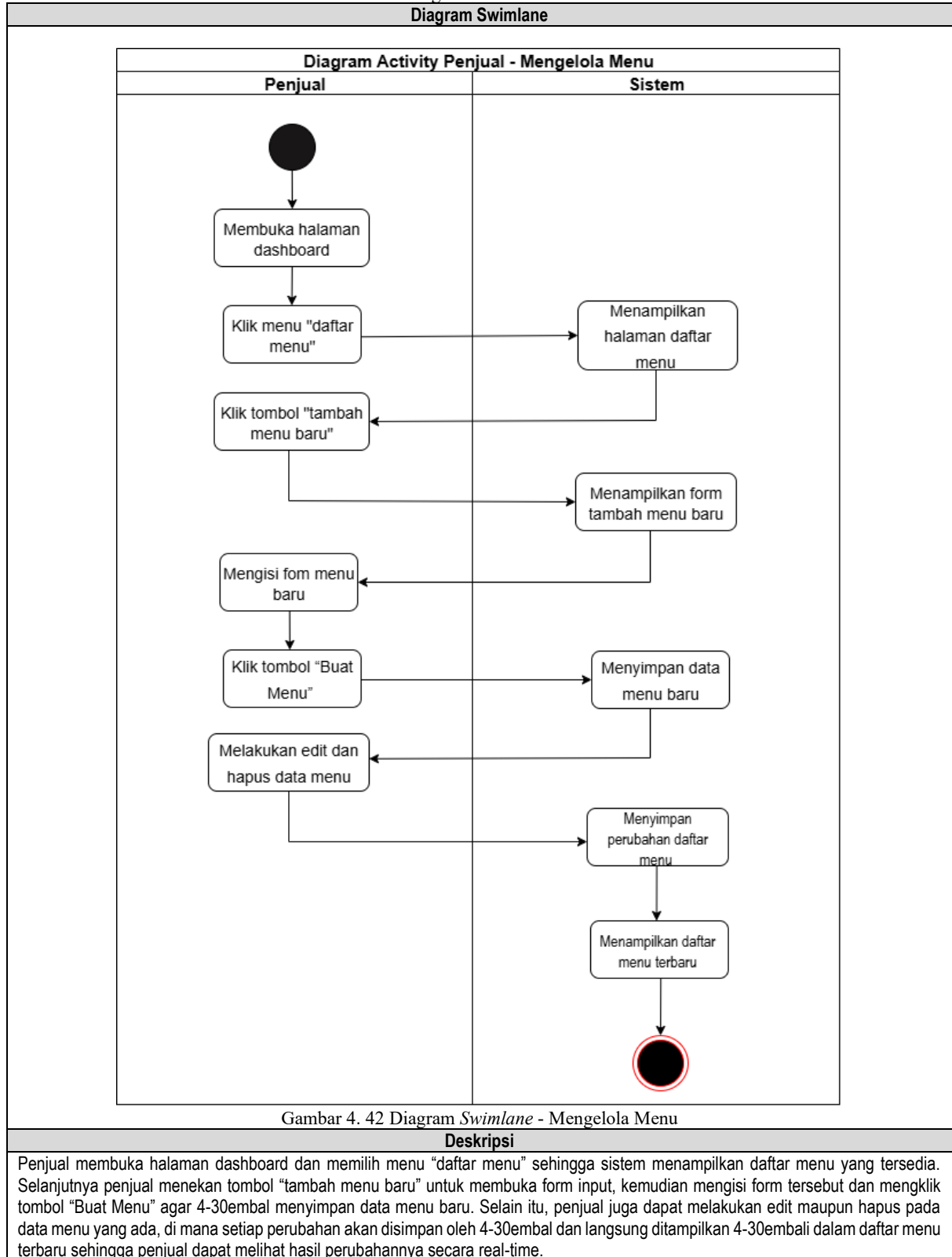
Tabel 4. 31 Diagram Swimlane UC-08



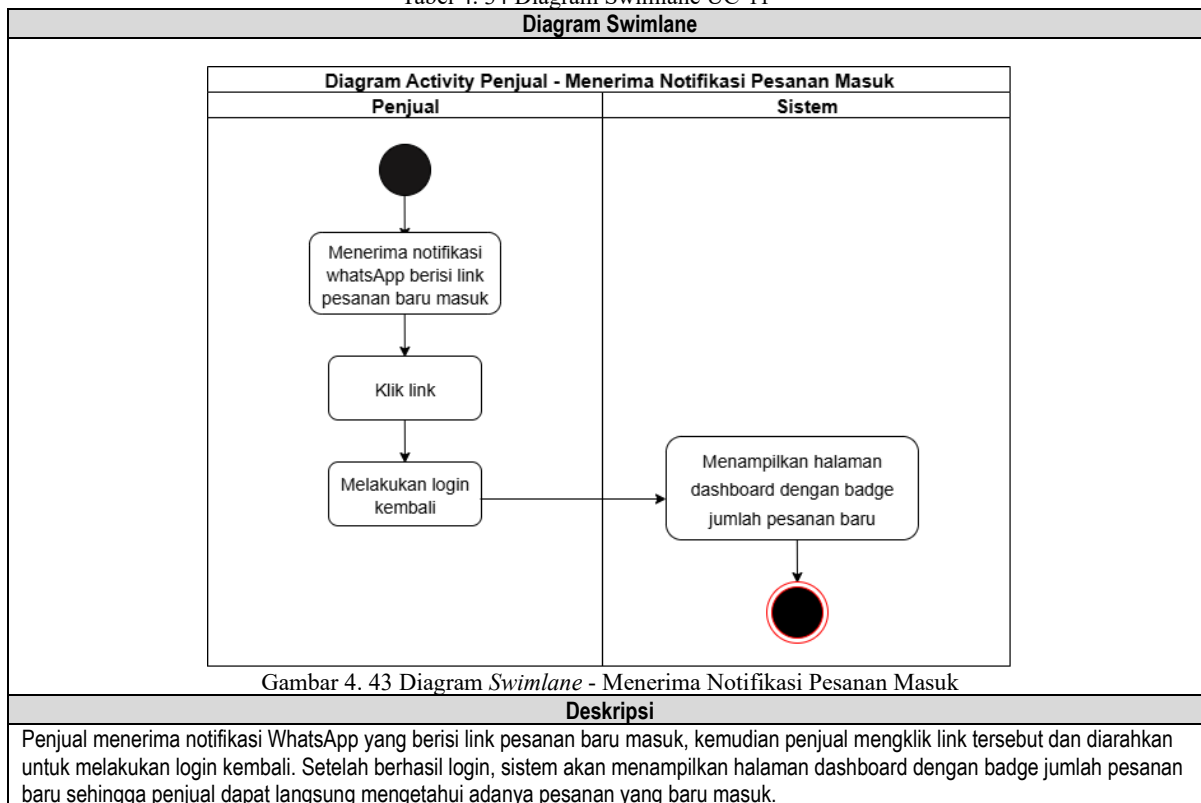
Tabel 4. 32 Diagram Swimlane UC-09



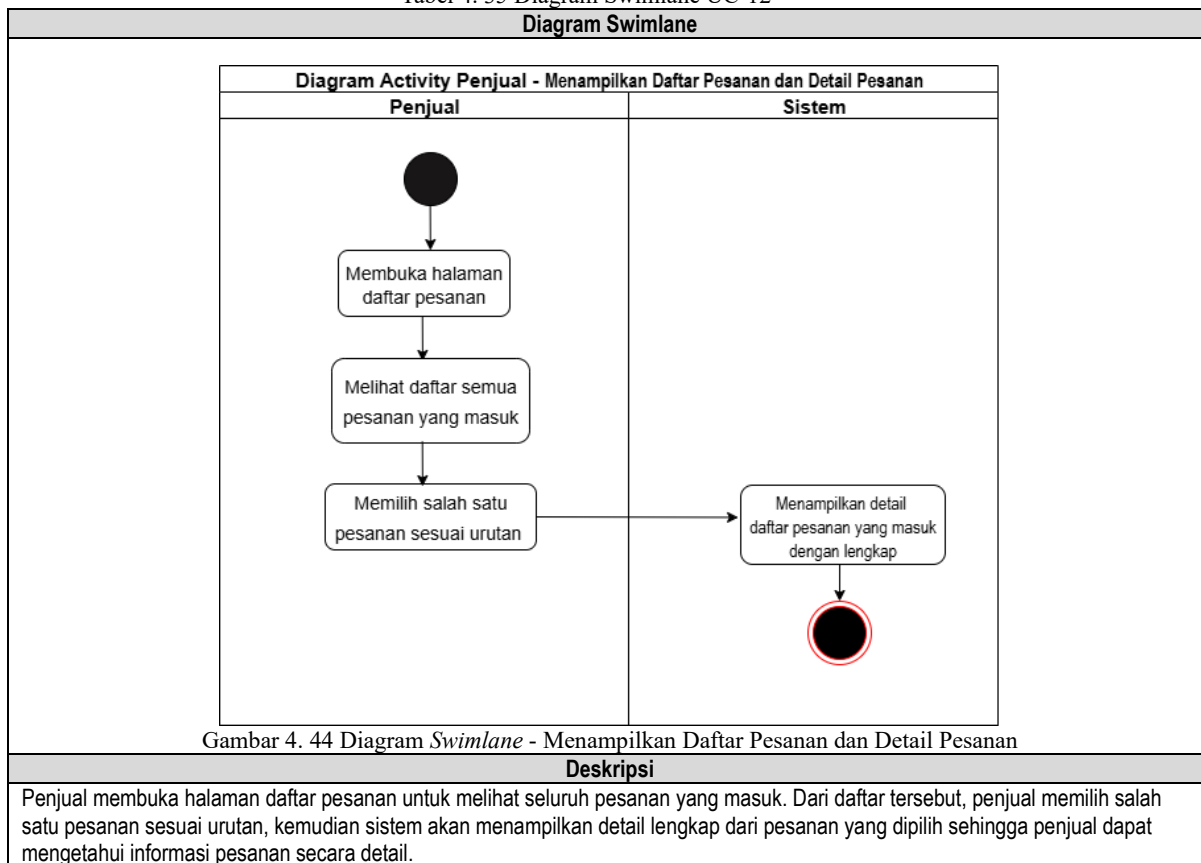
Tabel 4. 33 Diagram Swimlane UC-10



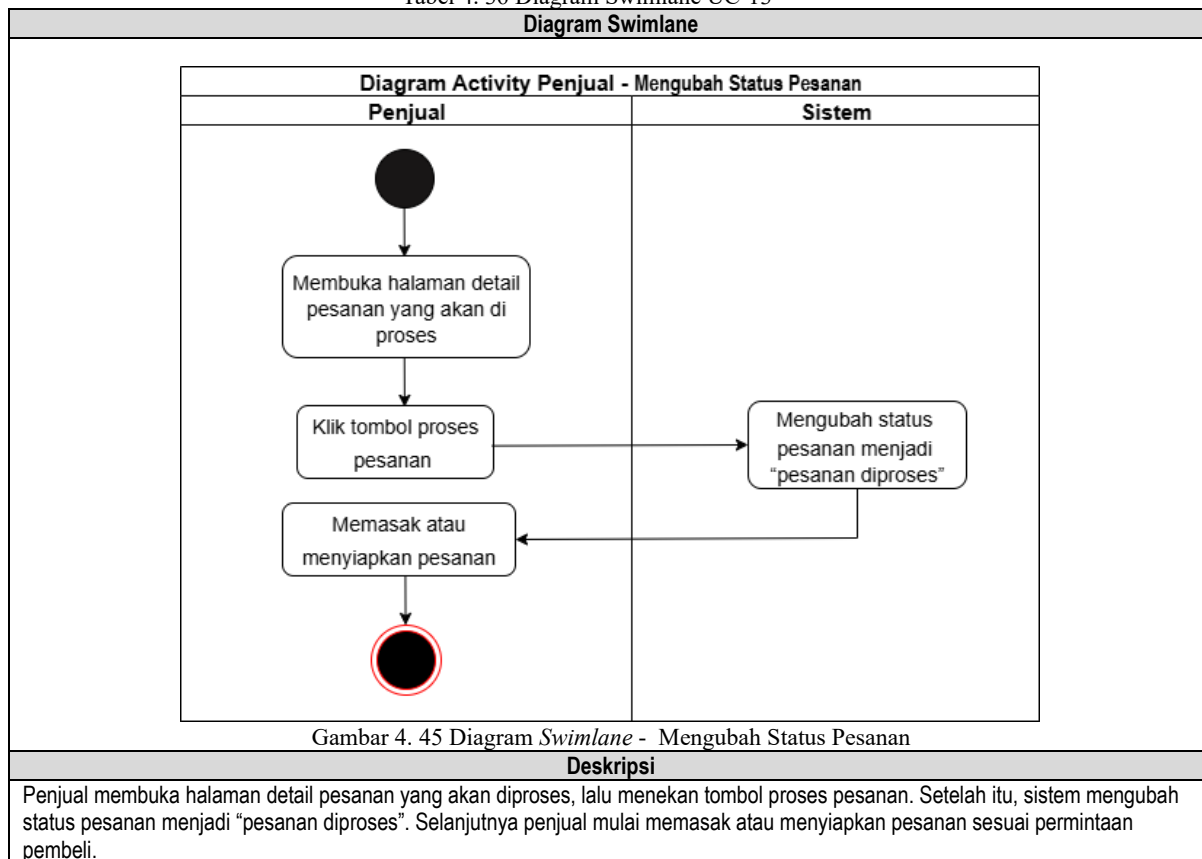
Tabel 4. 34 Diagram Swimlane UC-11



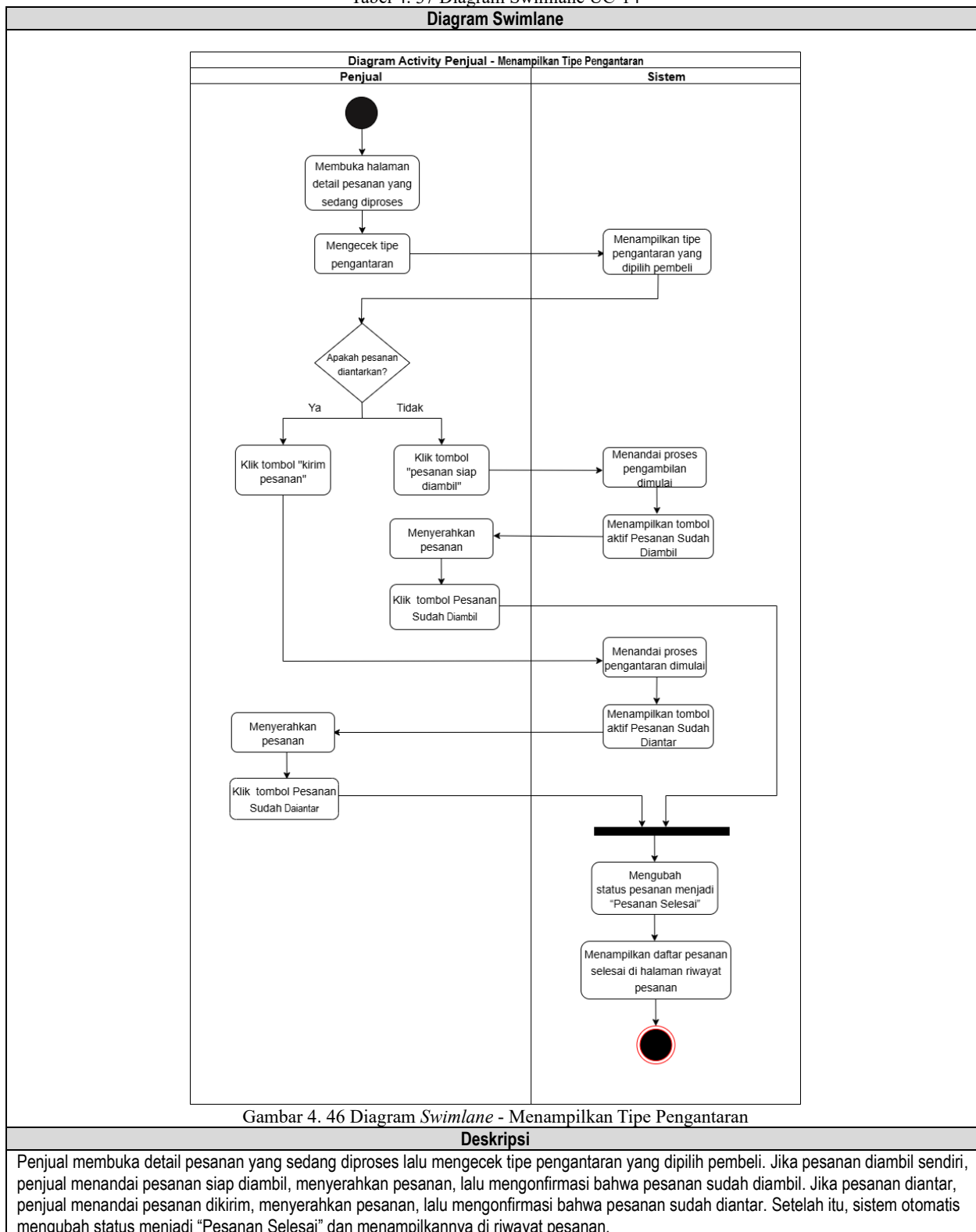
Tabel 4. 35 Diagram Swimlane UC-12



Tabel 4. 36 Diagram Swimlane UC-13



Tabel 4. 37 Diagram Swimlane UC-14



4.3 Perancangan Kebutuhan Perangkat Lunak

Tahap perancangan perangkat lunak bertujuan untuk mengubah hasil analisis kebutuhan menjadi rancangan teknis yang lebih terperinci dan sistematis. Pada tahap ini disusun gambaran arsitektur sistem, pemodel data, pemodel berbasis kelas, serta pemodel perilaku yang akan dijadikan pedoman utama dalam proses pembangunan aplikasi.

4.3.1 Pemodelan Data

Kebutuhan perangkat lunak juga mencakup aspek yang berkaitan dengan pembuatan, pengembangan, maupun integrasi dengan basis data. Oleh karena itu, perlu ditentukan secara jelas objek-objek data yang akan dikelola dalam sistem, hubungan antar objek tersebut, serta informasi tambahan yang berhubungan dengan relasi yang terbentuk.

4.3.1.1 Objek Data

Objek data adalah struktur yang merepresentasikan data tertentu di dalam perangkat lunak, biasanya berbentuk state atau props pada komponen *front-end*, yang nilainya berasal dari *back-end*.

Tabel 4. 38 Objek Data

No	Nama Objek	Deskripsi
1	Penjual	Menampilkan informasi penjual yang sedang login, seperti nama, email, dan nomor WhatsApp.
2	Kios	Menampilkan informasi kios yang dimiliki oleh penjual.
3	Menu	Menampilkan daftar menu dan detail yang tersedia di kios.
4	Pesanan	Menampilkan daftar pesanan dan riwayat pesanan pembeli, termasuk detail pesanan yang sedang berjalan serta detail riwayat pesanan yang telah selesai.
5	Keranjang	Menampilkan daftar menu yang ditambahkan ke keranjang oleh pembeli.
6	Pesanan Detail	Menampilkan rincian item di dalam satu pesanan.

4.3.1.2 Atribut data

Atribut data merupakan elemen yang mendeskripsikan karakteristik dari suatu objek data pada *front-end*. Atribut ini biasanya diwujudkan dalam bentuk state atau props, yang nilainya diperoleh dari API *back-end* atau didefinisikan secara lokal di dalam aplikasi. Dengan adanya atribut data, setiap komponen dapat mengelola struktur informasi yang dibutuhkan untuk ditampilkan kepada pengguna dan mendukung proses interaksi di antarmuka.

Tabel 4. 39 Atribut Data

No	Nama	Atribut	Type	Contoh Value
1	Penjual	id	string	{id-acak}
		nama	string	Syifa
		no_hp	string	085140702089
		email	string	syifa@gmail.com
		password	string	Syifa123
2	Kios	id	string	{id-acak}
		penjual_id	string	{id-acak}
		nama_kios	string	Dapur Uje
		nama_bank	string	BNI

No	Nama	Atribut	Type	Contoh Value
3	Menu	nomor_rekening	string	0512563456
		deskripsi	string	Ayam dan mie hot plate
		gambar_kios	string	/images/toko.jpg
		id	string	{id-acak}
		penjual_id	string	{id-acak}
		kios_id	string	{id-acak}
		nama_menu	string	Nasi Katsu
		deskripsi	string	Nasi dengan ayam katsu
		harga	price	Rp 15.000
		foto_menu	string	/images/katsu.jpg
		status_tersedia	string	Tersedia
		estimasi_menit	string	10
4	Pesanan	id	string	{id-acak}
		nama_pemesan	string	Legi
		total_harga	number	20000
		catatan	string	Ngga pake sambal
		guest_id	string	{id-acak}
		tipe_pengantaran	string	Ambil sendiri
		no_hp	string	085140702089
		diantar_ke	string	Depan perpus
		status	string	Menunggu Diproses
		total_estimasi	number	5
		kios_id	string	{id-acak}
5	Keranjang	id	string	{id-acak}
		guest_id	string	{id-acak}
		menu_id	string	{id-acak}
		jumlah	number	3
		total_harga	number	20000
		kios_id	string	{id-acak}
6	Pesanan Detail	id	string	{id-acak}
		pesanan_id	string	{id-acak}
		menu_id	string	{id-acak}
		nama_menu	string	Nasi Katsu
		jumlah	number	2
		harga	price	Rp 15.000
		subtotal	number	30000
		foto_menu	string	/images/katsu.jpg

4.3.2 Pemodelan Berbasis Kelas

Pemodelan berbasis kelas adalah cara untuk menggambarkan struktur perangkat lunak dengan menampilkan kelas-kelas yang merepresentasikan objek dalam sistem, atribut serta operasi yang dimilikinya, hubungan antar kelas, dan interaksi atau kolaborasi yang terjadi di antara kelas-kelas tersebut.

4.3.2.1 Identifikasi Kelas analisis

Identifikasi kelas analisis dilakukan untuk menemukan dan mendefinisikan kelas-kelas utama yang relevan dengan kebutuhan sistem. Proses ini didasarkan pada hasil analisis kebutuhan yang telah dituangkan dalam model *use case*, sehingga setiap kelas yang dihasilkan dapat merepresentasikan objek, atribut, serta tanggung jawab yang diperlukan oleh perangkat lunak. Berikut adalah kelas-kelas analisis yang berhasil diidentifikasi dari tahapan tersebut:

Tabel 4. 40 Identifikasi Kelas Analisis

No	Nama Kelas	Jenis Kelas	Keterangan
1	BerandaView	Boundary	Halaman beranda menampilkan 5 menu terbaru dan 8 kios terbaru, serta mencari kios atau menu
2	KiosView	Boundary	Halaman daftar kios yang menampilkan semua daftar kios dengan pencarian.
3	MenuView	Boundary	Halaman daftar menu yang menampilkan semua daftar menu perkios dengan pencarian.
4	DetailMenuView	Boundary	Halaman detail menu yang menampilkan informasi lengkap gambar, nama menu, harga, dan deskripsi menu.
5	KeranjangView	Boundary	Sidebar yang muncul untuk menampilkan isi keranjang dan total harga, dengan opsi ubah jumlah atau hapus.
6	KonfirmasiPesananView	Boundary	Halaman konfirmasi pesanan, termasuk memilih tipe pengantaran, nama, no WhatsApp, catatan tambahan, diantar ke?, dan metode pembayaran.
7	StatusPesananView	Boundary	Halaman status pesanan yang menampilkan detail pesanan, status pesanan, dan estimasi waktu penyelesaian pesanan.
8	DashboardView	Boundary	Halaman dashboard yang menampilkan total pesanan masuk, total menu, dan total pendapatan.
9	MenuDashboardView	Boundary	Menampilkan daftar menu di dashboard
10	TambahMenuDashboardView	Boundary	Halaman untuk menambahkan menu
11	UbahMenuDashboardView	Boundary	Halaman untuk mengubah data menu
12	DetailMenuDashboardView	Boundary	Bottom Sheet yang menampilkan detail menu (nama menu, harga, deskripsi, waktu penyajian)
13	PesananDashboardView	Boundary	Menampilkan daftar pesanan masuk di dashboard
14	DetailPesananDashboardView	Boundary	Halaman untuk melihat detail dan mengelola pesanan.
15	RiwayatPesananDashboardView	Boundary	Halaman untuk melihat Riwayat pesanan pembeli
16	DetailRiwayatPesananDashboardView	Boundary	Halaman yang menampilkan detail dari Riwayat pesanan.
17	ProfilDashboardView	Boundary	Halaman profile untuk melihat & mengubah data diri dan data toko
18	LoginView	Boundary	Halaman login penjual.
19	RegisterView	Boundary	Halaman registrasi akun penjual
20	RegisterTokoView	Boundary	Halaman registrasi kios (nama kios, bank, nomor rekening).
21	OtpView	Boundary	Halaman input & verifikasi OTP saat registrasi
22	ForgotPasswordView	Boundary	Halaman untuk meminta tautan reset kata sandi
23	NewPasswordView	Boundary	Halaman untuk membuat kata sandi baru.
24	KiosController	Controller	Mengelola logika bisnis yang berkaitan dengan kios, seperti memproses permintaan untuk menampilkan daftar kios, mengambil detail kios, dan mengelola data kios melalui API.
25	MenuController	Controller	Mengelola logika bisnis yang berkaitan dengan menu, seperti memproses permintaan untuk menampilkan daftar menu per kios, mengambil detail menu, menambahkan menu baru, mengubah data menu, serta menghapus data menu melalui API.
26	PenjualController	Controller	Mengelola logika bisnis yang berhubungan dengan penjual, mencakup proses autentikasi (login, register, verifikasi), serta pengelolaan data profil penjual melalui API.
27	PesananController	Controller	Mengelola logika bisnis pesanan, mulai dari membuat pesanan baru, menampilkan daftar dan detail pesanan, hingga memperbarui status pesanan serta memproses informasi pembayaran melalui API.
28	AuthController	Controller	Mengelola logika bisnis yang berhubungan dengan autentikasi, seperti memproses login, register, verifikasi OTP, lupa kata sandi, serta reset kata sandi dengan memanfaatkan API..
29	KeranjangController	Controller	Mengelola logika bisnis untuk keranjang belanja, seperti menambahkan menu ke keranjang, memperbarui jumlah item, menghapus item, serta menghitung total harga melalui API.
30	PaymentController	Controller	Mengelola logika bisnis yang berkaitan dengan pembayaran, seperti memproses permintaan untuk melakukan inisiasi transaksi, mengecek

No	Nama Kelas	Jenis Kelas	Keterangan
			status pembayaran yang sedang berlangsung, serta memperbarui hasil transaksi di <i>front-end</i> berdasarkan respons API.
31	DashboardController	Controller	Mengelola logika bisnis yang berkaitan dengan ringkasan dashboard penjual, seperti memproses permintaan untuk menampilkan total pesanan masuk, total menu yang tersedia, serta total pendapatan melalui API.
32	Penjual	Entity	Merepresentasikan struktur data akun penjual, mencakup informasi login (nama, email, nomor WhatsApp) dan profil penjual.
33	Kios	Entity	Merepresentasikan struktur data kios milik penjual, termasuk nama kios, deskripsi, gambar, nomor rekening, nama bank.
34	Menu	Entity	Merepresentasikan data menu di dalam kios, baik dalam bentuk daftar ringkas untuk ditampilkan maupun detail lengkap (nama, harga, deskripsi, gambar, status ketersediaan, estimasi penyajian).
35	Pesanan	Entity	Merepresentasikan struktur data untuk satu pesanan, mencakup identitas pemesan, status, total pembayaran, metode pengantaran, dan rincian item pesanan (riwayat maupun yang sedang berjalan).
36	Keranjang	Entity	Merepresentasikan keseluruhan data keranjang belanja, yang berisi daftar item menu yang ditambahkan pembeli, termasuk jumlah, dan total harga sebelum checkout.
37	Pesanan Detail	Entity	Merepresentasikan detail per-item dalam satu pesanan, mencakup referensi ke pesanan dan menu, nama menu, jumlah, harga, total harga, serta foto menu.

4.3.2.2 Identifikasi Atribut

Atribut adalah elemen data yang melekat pada suatu objek atau kelas dan digunakan untuk menggambarkan informasi tertentu yang relevan. Pada sisi *front-end*, atribut biasanya direpresentasikan melalui state, props, atau variabel yang menyimpan nilai untuk ditampilkan pada antarmuka, dikirim ke *back-end*, maupun dikelola sebagai hasil respon API. Keberadaan atribut membuat komponen dapat menyajikan data secara dinamis sekaligus mendukung interaksi pengguna dalam aplikasi.

Tabel 4. 41 Identifikasi Atribut

No	Nama Kelas	Atribut	Deskripsi
1	BerandaView	nama_kios	Menampilkan nama kios pada kartu kios di halaman beranda.
		Deskripsi	Menampilkan deskripsi kios untuk memberikan informasi kepada pengguna.
		gambar_kios	Menampilkan URL gambar kios yang akan ditampilkan sebagai gambar pada kartu kios.
		nama_menu	Menampilkan nama menu terbaru pada slider menu di beranda.
		Harga	Menampilkan harga menu terbaru dalam format rupiah di slider menu.
		foto_menu	Menampilkan URL gambar menu terbaru untuk ditampilkan pada slider di beranda.
2	KiosView	nama_kios	Menampilkan nama kios pada daftar kios.
		Deskripsi	Menampilkan deskripsi dari kios.
		gambar_kios	Menampilkan URL gambar kios untuk ditampilkan pada kartu kios.
3	MenuView	nama_menu	Menyimpan dan menampilkan nama menu yang ada pada kios.
		Harga	Menampilkan harga menu.
		foto_menu	Menyimpan URL gambar menu untuk ditampilkan di kartu menu.
		status_tersedia	Menampilkan status ketersediaan menu (Tersedia/Tidak Tersedia).
4	DetailMenuView	nama_menu	Menampilkan nama menu sebagai judul utama.

No	Nama Kelas	Atribut	Deskripsi
		foto_menu	Menampilkan URL gambar menu untuk ditampilkan sebagai gambar utama pada halaman detail.
		Deskripsi	Menampilkan deskripsi lengkap menu.
		Harga	Menampilkan harga menu dalam format rupiah.
5	KeranjangView	nama_menu	Menampilkan nama menu yang ditambahkan ke keranjang.
		foto_menu	Menampilkan gambar menu di dalam keranjang.
		Jumlah	Menyimpan dan menampilkan jumlah item menu di keranjang.
		Harga	Menampilkan harga per item menu di keranjang.
		total_harga	Menyimpan dan menampilkan total harga seluruh item dalam keranjang.
6	KonfirmasiPesananView	nama_kios	Menampilkan nama kios pada Ringkasan Pesanan.
		tipe_pengantaran	Menampilkan metode pengantaran yang dipilih (Pesan Antar / Ambil Sendiri).
		foto_menu	Menampilkan URL gambar menu item dalam ringkasan pesanan.
		nama_menu	Menampilkan nama menu item dalam ringkasan pesanan.
		Harga	Menampilkan harga per menu.
		Jumlah	Menampilkan jumlah tiap item menu.
		total_qty	Menampilkan jumlah total item dalam pesanan.
		total_harga	Menampilkan total harga semua item dalam pesanan.
		nama_pemesan	Menyimpan dan menampilkan input nama pembeli di formulir konfirmasi.
		no_hp	Menyimpan dan menampilkan input nomor WhatsApp pembeli di formulir konfirmasi.
		catatan	Menyimpan dan menampilkan input catatan tambahan dari pembeli.
7	StatusPesananView	Status	Menampilkan status pesanan (Menunggu Diproses / Sedang Diproses / Selesai).
		Progress	Menampilkan progres visual alur pesanan.
		total_estimasi	Menampilkan total estimasi waktu pesanan
		tipe_pengantaran	Menampilkan metode pengantaran (Pesan Antar / Ambil Sendiri).
		nama_kios	Menampilkan nama kios pesanan.
		nama_menu	Menampilkan nama menu pada daftar item pesanan.
		foto_menu	Menampilkan URL gambar menu pesanan.
		Jumlah	Menampilkan jumlah kuantitas tiap item menu.
		Harga	Menampilkan subtotal harga tiap item.
		nama_pemesan	Menampilkan nama pembeli
		no_hp	Menampilkan nomor WhatsApp pembeli
8	DashboardView	Catatan	Menampilkan catatan tambahan
		diantar_ke	Menampilkan alamat pengantaran bila tipe = Pesan Antar.
9	MenuDashboardView	total_pesanan	Menampilkan jumlah total pesanan masuk.
		total_menu	Menampilkan jumlah total menu yang tersedia.
		total_pendapatan	Menampilkan jumlah total pendapatan penjual.
		nama_menu	Menampilkan nama menu.
		Harga	Menampilkan harga menu.
		foto_menu	Menampilkan URL gambar menu sebagai

No	Nama Kelas	Atribut	Deskripsi
		status_tersedia	Menampilkan status menu (Tersedia/Tidak Tersedia)
10	TambahMenuDashboardView	nama_menu	Menyimpan dan menampilkan input nama menu baru.
		Harga	Menyimpan dan menampilkan input harga menu baru.
		foto_menu	Menyimpan URL gambar menu baru yang diunggah.
		Deskripsi	Menyimpan dan menampilkan input deskripsi menu baru.
		estimasi_menit	Menyimpan dan menampilkan estimasi waktu penyajian menu.
11	UbahMenuDashboardView	nama_menu	Menyimpan dan menampilkan nama menu untuk diubah.
		Harga	Menyimpan dan menampilkan harga menu untuk diubah.
		foto_menu	Menampilkan / mengubah gambar menu.
		deskripsi	Menyimpan dan menampilkan deskripsi menu untuk diubah.
		estimasi_menit	Menyimpan dan menampilkan estimasi penyajian untuk diubah.
12	DetailMenuDashboardView	nama_menu	Menampilkan nama menu di detail menu.
		Harga	Menampilkan harga menu.
		Deskripsi	Menampilkan deskripsi menu.
		estimasi_menit	Menampilkan estimasi penyajian menu.
13	PesananDashboardView	Nomor	Menampilkan nomorurut/antrian
		nama_pemesan	Menampilkan nama pelanggan
		tanggal_bayar	Menampilkan tanggal
		total_harga	Menampilkan total harga dalam format rupiah.
		Status	Menampilkan status pesanan.
14	DetailPesananDashboardView	Status	Menyimpan dan menampilkan status pesanan
		nomor_antrian	Menampilkan nomor antrian pesanan
		nama_pemesan	Menampilkan nama pelanggan yang membuat pesanan.
		tanggal_bayar	Menampilkan tanggal/waktu pesanan
		tipe_pengantaran	Menampilkan metode pengantaran
		metode_bayar	Menampilkan metode pembayaran
		no_hp	Menampilkan nomor WhatsApp pelanggan.
		nama_menu	Menampilkan nama setiap menu dalam rincian pesanan.
		Jumlah	Menampilkan jumlah item per menu.
		Harga	Menampilkan subtotal item.
		Catatan	Menampilkan catatan tambahan dari pelanggan.
		total_harga	Menampilkan total harga pesanan dalam format rupiah.
		Timer	Menampilkan countdown waktu tersisa bila status sedang diproses.
15	RiwayatPesananDashboardView	Nomor	Menampilkan nomorurut/antrian dalam daftar riwayat.
		nama_pemesan	Menampilkan nama pelanggan
		tanggal_bayar	Menampilkan tanggal
		total_harga	Menampilkan total harga dalam format rupiah.
		Status	Menampilkan status terakhir pesanan.
16	DetailRiwayatPesananDashboardView	Status	Menampilkan status terakhir pesanan
		nomor_antrian	Menampilkan nomor antrian pesanan
		nama_pemesan	Menampilkan nama pelanggan yang membuat pesanan.
		tanggal_bayar	Menampilkan tanggal/waktu pesanan
		tipe_pengantaran	Menampilkan metode pengantaran
		metode_bayar	Menampilkan metode pembayaran

No	Nama Kelas	Atribut	Deskripsi
		no_hp	Menampilkan nomor WhatsApp pelanggan.
		nama_menu	Menampilkan nama setiap menu dalam rincian pesanan.
		Jumlah	Menampilkan jumlah item per menu.
		Harga	Menampilkan subtotal item.
		Catatan	Menampilkan catatan tambahan dari pelanggan.
		total_harga	Menampilkan total harga pesanan dalam format rupiah.
17	ProfilDashboardView	nama_penjual	Menyimpan dan menampilkan nama penjual.
		no_hp	Menyimpan dan menampilkan nomor WhatsApp penjual.
		Email	Menyimpan dan menampilkan email penjual.
		gambar_kios	Menyimpan dan menampilkan URL gambar kios.
		nama_kios	Menyimpan dan menampilkan nama kios.
		Deskripsi	Menyimpan dan menampilkan deskripsi kios.
		nama_bank	Menyimpan dan menampilkan nama bank kios.
		nomor_rekening	Menyimpan dan menampilkan nomor rekening kios.
18	LoginView	nama_penjual	Menyimpan dan menampilkan input nama penjual di formulir login.
		Password	Menyimpan dan menampilkan input password di formulir login.
19	RegisterView	nama_penjual	Menyimpan dan menampilkan input nama penjual di formulir register.
		no_hp	Menyimpan dan menampilkan input nomor WhatsApp formulir register.
		Email	Menyimpan dan menampilkan input email di formulir register.
		Password	Menyimpan dan menampilkan input kata sandi di formulir register.
		confirm_password	Menyimpan dan menampilkan input konfirmasi kata sandi di formulir register.
20	RegisterTokoView	nama_kios	Menyimpan dan menampilkan input nama kios di formulir register toko.
		nama_bank	Menyimpan dan menampilkan input nama bank di formulir register toko.
		nomor_rekening	Menyimpan dan menampilkan input nomor rekening di formulir register toko.
21	OtpView	Otp	Menyimpan dan menampilkan input kode OTP untuk verifikasi akun.
22	ForgotPasswordView	No_hp	Menyimpan dan menampilkan input nomor WhatsApp penjual untuk reset kata sandi.
23	NewPasswordView	Password	Menyimpan dan menampilkan input kata sandi baru.
		confirm_password	Menyimpan dan menampilkan input konfirmasi kata sandi baru.
24	KiosController	Create	Mengirim request POST ke API untuk mendaftarkan kios baru saat proses registrasi toko.
		getAll	Mengirim request GET ke API untuk mengambil daftar semua kios.
		getHomepage	Mengirim request GET ke API untuk mengambil 8 kios terbaru dan menampilkannya di halaman beranda pembeli.
		getMenusByKios	Mengirim request GET ke API untuk mengambil daftar menu berdasarkan kiosId.
		getById	Mengirim request GET ke API untuk mengambil detail kios tertentu berdasarkan id.
		searchAll	Mengirim request GET ke API untuk mencari data kios dan menu sekaligus.

No	Nama Kelas	Atribut	Deskripsi
		getKios	Mengirim request GET ke API untuk mengambil profil kios milik penjual yang sedang login.
		updateKios	Mengirim request PUT ke API untuk memperbarui data kios (nama kios, deskripsi, bank, nomor rekening, dan gambar kios).
25	MenuController	getNewMenus	Mengirim request GET ke API untuk mengambil daftar menu terbaru
		getMenuByld	Mengirim request GET ke API untuk mengambil detail menu versi pembeli.
		getMenus	Mengirim request GET ke API untuk mengambil semua menu milik penjual.
		getMenuByld	Mengirim request GET ke API untuk mengambil detail satu menu milik penjual.
		addMenu	Mengirim request POST ke API untuk menambahkan menu baru.
		updateMenu	Mengirim request PUT ke API untuk memperbarui menu.
		deleteMenu	Mengirim request DELETE ke API untuk menghapus menu.
		getMenusPaginated	Mengirim request GET ke API untuk mengambil daftar menu dengan paginasi.
26	PenjualController	Create	Mengirim request POST ke API untuk mendaftarkan akun penjual saat proses registrasi.
		getProfile	Mengirim request GET ke API untuk mengambil data profil penjual yang sedang login.
		updateProfile	Mengirim request PUT ke API untuk memperbarui profil penjual.
27	PesananController	buatPesanan	Mengirim request POST ke API untuk membuat pesanan baru berdasarkan item di keranjang (pembeli).
		getPesananDetail	Mengirim request GET API untuk mengambil detail lengkap pesanan tertentu (pembeli).
		getStatusPesananGuest	Mengirim request GET ke API untuk memeriksa status pesanan milik pembeli (menunggu, diproses, selesai).
		getPesananMasuk	Mengirim request GET API untuk mengambil daftar pesanan masuk (penjual).
		getDetailPesananMasuk	Mengirim request GET ke API untuk mengambil detail satu pesanan masuk.
		getCountPesananMasuk	Mengirim request GET API untuk menghitung jumlah pesanan masuk (penjual).
		updateStatusPesanan	Mengirim request PATCH ke API untuk memperbarui status pesanan (menunggu, diproses, selesai).
		getRiwayatPesanan	Mengirim request GET ke API untuk mengambil daftar riwayat pesanan (penjual).
		getDetailRiwayatPesanan	Mengirim request GET ke API untuk mengambil detail dari riwayat pesanan tertentu (penjual).
28	AuthController	Register	Mengirim request POST ke API untuk mendaftarkan akun baru.
		registerKios	Mengirim request POST ke API untuk mendaftarkan kios baru.
		verifyOtp	Mengirim request POST ke API untuk memverifikasi kode OTP yang dikirim via WhatsApp.
		resendOtp	Mengirim request POST ke API untuk meminta ulang OTP ke nomor WhatsApp penjual.

No	Nama Kelas	Atribut	Deskripsi
		Login	Mengirim request POST ke API untuk autentikasi pengguna dengan nama dan password (opsional rememberMe).
		forgotPassword	Mengirim request POST ke API untuk memulai proses reset password menggunakan nomor WhatsApp.
		resetPassword	Mengirim request POST ke API untuk mengatur ulang password menggunakan token valid beserta password dan confirmPassword baru.
		Logout	Mengirim request POST ke API untuk mengakhiri sesi pengguna.
29	KeranjangController	getKeranjang	Mengirim request GET ke API untuk mengambil daftar item dalam keranjang berdasarkan guest_id.
		addItem	Mengirim request POST ke API untuk menambahkan menu baru ke dalam keranjang.
		updateItem	Mengirim request PUT ke API untuk memperbarui jumlah dari item yang ada di keranjang berdasarkan guest_id.
		removeItem	Mengirim request DELETE ke API untuk menghapus item tertentu dari keranjang.
30	PaymentController	buatTransaksiMidtrans	Mengirim request POST ke API untuk membuat transaksi pembayaran baru melalui Midtrans.
31	DashboardController	getDashboardData	Mengirim request GET ke API untuk mengambil data ringkasan dashboard.
32	Penjual	Id	Atribut untuk identifikasi unik penjual di dalam aplikasi.
		Nama	Atribut menyimpan dan menampilkan nama lengkap penjual.
		no_hp	Atribut menyimpan nomor WhatsApp penjual untuk kontak dan verifikasi.
		Email	Atribut menyimpan alamat email penjual.
		Password	Atribut virtual yang merepresentasikan input kata sandi dari penjual pada form, tidak disimpan dalam state aplikasi.
		Otp	Merepresentasikan input kode OTP dari penjual pada form verifikasi.
33	Kios	Id	Atribut untuk identifikasi unik untuk setiap kios.
		penjual_id	Atribut yang menghubungkan kios dengan penjual yang memilikinya.
		nama_kios	Atribut untuk menyimpan nama kios.
		nama_bank	Atribut untuk menyimpan nama bank
		nomor_rekening	Atribut untuk menyimpan nomor rekening
		Deskripsi	Atribut untuk menyimpan deskripsi kios.
		gambar_kios	Atribut menyimpan URL gambar kios.
34	Menu	Id	Atribut untuk identifikasi unik setiap menu.
		penjual_id	Atribut yang menghubungkan menu dengan penjual.
		kios_id	Atribut untuk menghubungkan menu dengan kios.
		nama_menu	Atribut untuk menyimpan dan menampilkan nama menu.
		Deskripsi	Atribut untuk menyimpan deskripsi detail menu.
		Harga	Atribut untuk menyimpan harga menu
		foto_menu	Atribut untuk menyimpan URL gambar menu.
		status_tersedia	Atribut untuk menyimpan status ketersediaan menu (tersedia/tidak tersedia).
35	Pesanan	estimasi_menit	Atribut untuk menyimpan estimasi waktu penyajian menu dalam menit.
		Id	Atribut untuk identifikasi unik setiap pesanan.

No	Nama Kelas	Atribut	Deskripsi
		nama_pemesan	Atribut untuk menyimpan nama pembeli yang melakukan pesanan.
		total_harga	Atribut untuk menyimpan total harga seluruh item dalam pesanan.
		Catatan	Atribut untuk menyimpan catatan tambahan dari pembeli.
		guest_id	Atribut untuk menghubungkan pesanan dengan pembeli.
		tipe_pengantaran	Atribut untuk menyimpan tipe pengantaran (ambil sendiri/pesan antar).
		no_hp	Atribut untuk menyimpan nomor WhatsApp pembeli.
		diantar_ke	Atribut untuk menyimpan lokasi tujuan pengantaran.
		Status	Atribut untuk menyimpan status terkini pesanan.
		total_estimasi	Atribut untuk menyimpan total estimasi waktu penyelesaian pesanan (menit).
		kios_id	Atribut untuk menghubungkan pesanan dengan kios terkait.
36	Keranjang	Id	Atribut untuk identifikasi unik setiap keranjang.
		guest_id	Atribut untuk menghubungkan keranjang dengan pembeli.
		menu_id	Atribut untuk menghubungkan item keranjang dengan menu yang dipilih.
		Jumlah	Atribut untuk menyimpan jumlah item menu yang dimasukkan.
		total_harga	Atribut untuk menyimpan total harga seluruh item di keranjang.
		kios_id	Atribut untuk menghubungkan keranjang dengan kios asal item.
37	Pesanan Detail	Id	Atribut untuk identifikasi unik setiap item pesanan.
		pesanan_id	Atribut untuk menghubungkan detail item dengan pesanan.
		menu_id	Atribut untuk menghubungkan detail item dengan menu yang dipesan.
		nama_menu	Atribut untuk menyimpan nama menu pada saat transaksi.
		Jumlah	Atribut untuk menyimpan jumlah item menu yang dipesan.
		Harga	Atribut untuk menyimpan harga satuan menu pada saat transaksi.
		Subtotal	Atribut untuk menyimpan total harga peritem.
		foto_menu	Atribut untuk menyimpan URL gambar menu.

4.3.2.3 Identifikasi Operasi

Operasi pada dasarnya merupakan fungsi atau tindakan yang menjelaskan perilaku dari suatu kelas dalam sistem. Melalui operasi, setiap objek dapat melakukan pengolahan data, berinteraksi dengan kelas lain, maupun memberikan layanan kepada pengguna. Tabel berikut menyajikan secara rinci daftar operasi yang dimiliki oleh masing-masing kelas yang telah diidentifikasi pada tahap sebelumnya.

Tabel 4. 42 Identifikasi Operasi

No	Nama Kelas	Kandidat Perilaku	Nama Perilaku
1	BerandaView	Mengambil dan menampilkan daftar semua kios atau berdasarkan pencarian.	fetchKios()
		Memperbarui state untuk teks pencarian.	setSearch(value)

No	Nama Kelas	Kandidat Perilaku	Nama Perilaku
		Menavigasi ke halaman daftar semua kios.	navigate("/KiosPage")
		Mengambil dan menampilkan daftar menu baru (slider).	fetchNewMenus()
		Menangani error jika gambar menu dan kios gagal dimuat dan menggantinya.	onError()
		Menavigasi pengguna ke halaman menu dari kios yang dipilih.	handleNavigate()
2	KiosView	Mengambil dan menampilkan daftar semua kios atau berdasarkan pencarian	fetchKios()
		Memperbarui state untuk teks pencarian.	setSearch(value)
		Menavigasi pengguna ke halaman menu dari kios yang dipilih.	handleNavigate()
		Menangani error jika gambar kios gagal dimuat.	onError()
3	MenuView	Mengambil data detail kios dan daftar menu dari kios tersebut.	fetchKiosAndMenus()
		Memperbarui state untuk teks pencarian menu.	setSearchTerm(value)
		Menambah item ke keranjang atau menaikkan kuantitasnya.	addToCart(id)
		Mengurangi kuantitas item atau menghapusnya dari keranjang.	decreaseQty(id)
		Menampilkan komponen keranjang (cart).	setShowCart(true)
		Menutup dan menyembunyikan notifikasi/alert.	setAlertMessage(null)
		Menavigasi ke halaman detail menu saat kartu diklik.	goToDetail()
		Mengelola perubahan kuantitas dari tombol +/-	handleSetQty(newQty)
		Menambahkan item ke keranjang saat tombol "Tambah Keranjang" diklik.	addToCart(id)
4	DetailMenuView	Mengambil data detail untuk satu menu.	fetchMenu()
		Mengelola semua operasi keranjang (tambah, ubah, hapus) untuk menu yang dipilih.	handleQuantityChange(newQty)
		Mengembalikan pengguna ke halaman sebelumnya.	handleGoBack()
		Menangani error jika gambar menu gagal dimuat.	handleImageError()
		Menampilkan atau menyembunyikan komponen keranjang.	setShowCart(boolean)
5	KeranjangView	Menutup keranjang dan mengarahkan pembeli ke halaman konfirmasi pesanan.	handleCheckout()
		Menutup tampilan keranjang.	onClose()
		Mendeteksi ukuran layar untuk menyesuaikan tampilan (desktop/mobile).	checkScreen()
		Memperbarui jumlah kuantitas item di keranjang.	updateQty(id, newQty)
		Menghapus item dari keranjang jika kuantitasnya menjadi nol.	removeItem(id)
		Menangani error jika gambar item gagal dimuat.	onError()
6	KonfirmasiPesananView	Memuat data keranjang belanja saat halaman dibuka.	loadCart()
		Mengambil detail kios berdasarkan item di keranjang.	fetchKios()
		Mengubah tipe pengantaran pesanan (Pesan Antar/Ambil Sendiri).	setDeliveryType(type)
		Mengarahkan pengguna kembali ke halaman menu untuk menambah item.	navigate('/MenuPage/\${kiosId}')
		Memperbarui atau menghapus kuantitas item di ringkasan pesanan.	updateQty() / removeItem()
		Memuat skrip gateway pembayaran (Midtrans Snap).	useEffect (for Snap.js)
		Memproses data form, membuat pesanan, dan memulai alur pembayaran.	handleSubmit(e)
		Membuka pop-up pembayaran dan menangani hasilnya (sukses, pending, gagal).	openSnap(token, id)
		Menampilkan notifikasi custom untuk status pembayaran atau error.	showNotification(...)
		Mengelola perubahan input pada setiap kolom form (nama pemesan, noHp, catatan, diantarke)	setNama(value), setNoHp(value), setCatatan(value), setDiantarKe(value)
7	StatusPesananView	Mengambil data detail & status pesanan.	fetchPesananData()
		Mengatur status pesanan menjadi "selesai".	handleOrderCompletion()
		Memperbarui tampilan progress bar berdasarkan waktu.	useEffect (for progress)
		Menangani saat timer estimasi waktu habis.	handleTimerFinish()
		Menutup notifikasi atau alert yang muncul.	handleCloseAlert()
		Mengatur ulang sesi untuk memulai pesanan baru.	handlePesanLagi()
		Membersihkan data sesi secara otomatis setelah pesanan selesai.	useEffect (for cleanup)
8	DashboardView	Mengambil data ringkasan (total pesanan, menu, dan pendapatan).	fetchDashboard()
9	MenuDashboardView	Mengambil daftar menu dengan paginasi.	loadMenus()
		Menampilkan notifikasi sukses.	showSuccess(msg)
		Menghapus data menu dari daftar.	handleDelete(id)

No	Nama Kelas	Kandidat Perilaku	Nama Perilaku
		Mengubah status ketersediaan stok menu (tersedia/tidak tersedia).	handleToggleStock(id, status)
		Berpindah halaman pada daftar menu.	handlePageChange(pageNumber)
		Menampilkan detail dari menu yang dipilih.	setSelectedMenu(menu)
		Mengarahkan pengguna ke halaman tambah menu baru.	navigate("/AddMenuPage")
10	TambahMenuDashboardView	Memproses dan mengirimkan data form untuk membuat menu baru.	handleSubmit(e)
		Mengelola file gambar yang diunggah dan menampilkan pratinjaunya.	handleImageUpload(file)
		Mengubah status ketersediaan menu melalui toggle switch.	setIsAvailable(boolean)
		Mengelola perubahan input pada setiap kolom form (nama, harga, deskripsi, dan waktu penyajian).	setMenuName(value), setPrice(value), setDescription(value), setServingTime(value)
11	UbahMenuDashboardView	Mengambil data detail menu yang akan diubah.	fetchMenuDetail()
		Memproses dan mengirimkan data form yang telah diubah.	handleUpdate(e)
		Mengelola file gambar baru yang diunggah dan menampilkan pratinjaunya.	handleImageUpload(file)
		Mengubah status ketersediaan menu melalui toggle switch.	setIsAvailable(boolean)
12	DetailMenuDashboardView	Mengelola perubahan input pada setiap kolom form (nama, harga, deskripsi, dan waktu penyajian).	setMenuName(value), setPrice(value), setDescription(value), setServingTime(value)
		Menutup tampilan detail menu.	onClose()
		Mengarahkan pengguna ke halaman untuk mengubah data menu.	handleEdit()
		Menghapus data menu dari daftar menu setelah konfirmasi.	handleDelete()
13	PesananDashboardView	Menampilkan atau menyembunyikan pop-up konfirmasi hapus.	setShowConfirm(boolean)
		Mengambil daftar pesanan (masuk atau riwayat) dengan paginasi.	fetchOrders(page)
		Berpindah halaman pada daftar pesanan.	setCurrentPage(pageNumber)
14	DetailPesananDashboardView	Mengarahkan pengguna ke halaman detail pesanan yang dipilih.	navigate('/OrderDetailPage/{type}/{row.id}')
		Menghitung dan menampilkan sisa waktu pesanan secara real-time.	hitungSisaWaktu()
		Menampilkan atau menyembunyikan detail tambahan info pelanggan (mobile).	setShowDetail(boolean)
		Mengambil data detail pesanan (masuk atau riwayat).	fetchOrderDetail()
		Mengirim pembaruan status pesanan.	updateStatus(newStatus)
		Mengelola alur perubahan status pesanan saat tombol utama diklik.	handleStatusChange()
		Mengonfirmasi dan menjalankan perubahan status dari pop-up.	handleConfirmSend()
15	RiwayatPesananDashboardView	Menampilkan atau menyembunyikan pop-up konfirmasi.	setShowConfirmModal(boolean)
		Mengambil data riwayat pesanan dengan paginasi.	Mengambil data riwayat pesanan dengan paginasi.
		Mengambil ulang data saat pengguna kembali ke tab browser.	handleFocus()
		Berpindah halaman pada daftar riwayat pesanan.	setCurrentPage(pageNumber)
16	DetailRiwayatPesananDashboardView	Mengarahkan pengguna ke halaman detail pesanan yang dipilih.	navigate('/OrderDetailPage/{type}/{order.id}', { state: { from: "history" } })
		Mengambil data detail riwayat pesanan.	fetchOrderDetail()
		Menampilkan atau menyembunyikan detail tambahan info pelanggan (mobile).	setShowDetail(boolean)
17	ProfilDashboardView	Mengambil data profil penjual.	loadProfile()
		Mengirimkan data pembaruan profil penjual.	handleSubmit(e)
		Membatalkan perubahan dan mengembalikan form ke data semula.	handleCancel()
		Mengelola perubahan input pada form (nama, no hp, email).	setName(value), setNoPhone(value), setEmail(value)
		Mengambil data profil toko.	fetchData()
		Mengirimkan data pembaruan profil toko.	handleSubmit(e)
		Mengelola unggah dan pratinjau gambar toko.	handleImageUpload(file)
		Membatalkan perubahan dan mengembalikan form ke data semula.	handleCancel()
		Mengelola perubahan input pada form (nama toko, deskripsi, nama bank, nomor rekening).	setStoreName(value), setDescription(value),

No	Nama Kelas	Kandidat Perilaku	Nama Perilaku
			setBankName(value), setAccountNumber(value)
		Mengganti tampilan antara tab "Edit Profil" dan "Edit Toko".	setActiveTab(tab)
		Menyimpan pilihan tab aktif ke localStorage.	useEffect (for activeTab)
18	LoginView	Mengambil nama pengguna yang tersimpan jika "Ingatkan Saya" pernah dipilih.	useEffect (for rememberMe)
		Mengirimkan data login dan menangani hasilnya (sukses/gagal).	handleSubmit(e)
		Menampilkan notifikasi sukses setelah login berhasil.	showSuccessAlert()
		Menampilkan notifikasi error jika login gagal.	showErrorAlert(message)
		Mengubah status centang pada opsi "Ingatkan Saya".	setRememberMe(boolean)
		Mengelola perubahan input pada form (nama dan kata sandi).	setNama(value), setPassword(value)
19	RegisterView	Memvalidasi semua input form sebelum pengiriman.	validate()
		Mengirimkan data registrasi dan menangani hasilnya.	handleSubmit(e)
		Mengelola perubahan input pada setiap kolom form.	setNama(value), setPhone(value), setEmail(value), setPassword(value), setConfirmPassword(value).
20	RegisterTokoView	Memproses dan mengirimkan data pendaftaran toko.	handleSubmit(e)
		Memvalidasi input nomor rekening secara real-time.	handleNoRekeningChange(e)
		Mengelola perubahan input pada form (nama toko, nama bank, nomor rekening).	setNamaToko(value), setNamaBank(value), setNoRekening(value).
21	OtpView	Menginisialisasi dan menjalankan timer hitung mundur untuk OTP.	useEffect (for timer)
		Mengirimkan kode OTP untuk verifikasi.	handleSubmit(e)
		Meminta pengiriman ulang kode OTP.	handleResend()
		Menampilkan notifikasi (sukses atau gagal).	showSuccessAlert(), showErrorAlert()
		Mengelola perubahan input pada kolom kode OTP.	setKodeOtp(value)
22	ForgotPasswordView	Mengirimkan permintaan reset kata sandi.	handleSubmit(e)
		Mengelola perubahan input pada kolom nomor telepon.	setPhone(value)
23	NewPasswordView	Memvalidasi input kata sandi baru dan konfirmasinya.	validate()
		Mengirimkan kata sandi baru untuk direset.	handleSubmit(e)
		Menampilkan notifikasi (sukses atau gagal).	showSuccessAlert(), showErrorAlert()
		Mengelola perubahan input pada form kata sandi.	setPassword(value), setConfirmPassword(value)
24	KiosController	Mengirim permintaan untuk mengambil data profil kios yang terhubung dengan akun penjual yang sedang login.	getKios(token)
		Mengirim permintaan untuk memperbarui data kios yang sudah ada.	updateKios(data, token, id)
		Mengirim permintaan untuk mengambil semua data kios. Jika ada searchTerm, fungsi akan melakukan pencarian.	getAll(searchTerm)
		Mengirim permintaan untuk mengambil daftar kios yang secara spesifik dipilih untuk tampilan halaman utama.	getHomepage()
		Mengirim permintaan untuk mengambil semua menu yang dimiliki oleh satu kios spesifik.	getMenusByKios(kiosId, searchTerm)
		Mengirim permintaan untuk mengambil informasi detail dari satu kios spesifik.	getById(kiosId)
		Mengirim permintaan untuk melakukan pencarian global yang mencakup data kios dan menu secara bersamaan.	searchAll(query)
25	MenuController	Mengirim permintaan untuk mengambil semua data menu yang terhubung dengan akun penjual yang sedang login.	getMenus(token)
		Mengirim permintaan untuk mengambil data detail satu menu spesifik dari sisi penjual (memerlukan login).	getMenuById(id, token)
		Mengirim permintaan untuk membuat data menu baru, termasuk mengunggah foto menu.	addMenu(formData, token)
		Mengirim permintaan untuk memperbarui data menu yang sudah ada.	updateMenu(id, data, token, isFormData)

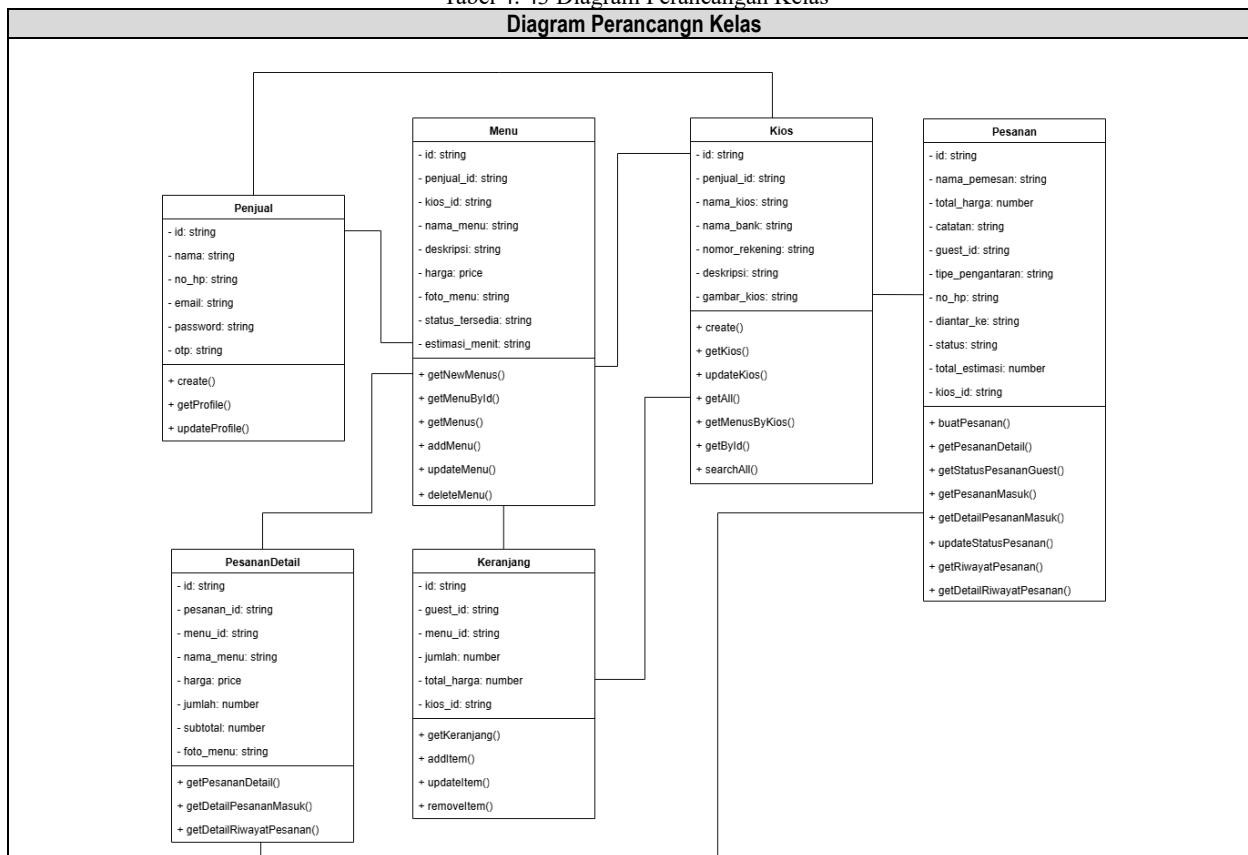
No	Nama Kelas	Kandidat Perilaku	Nama Perilaku
		Mengirim permintaan untuk menghapus data menu.	deleteMenu(id, token)
		Mengirim permintaan untuk mengambil daftar menu dengan sistem paginasi (dibagi per halaman).	getMenusPaginated(token, page, limit)
		Mengirim permintaan untuk mengambil daftar menu-menu terbaru yang bisa dilihat oleh pembeli.	getNewMenus()
		Mengirim permintaan untuk mengambil data detail satu menu spesifik dari sisi pembeli (tidak perlu login)	getMenuById(id)
26	PenjualController	Mengirim permintaan untuk mengambil data profil penjual yang sedang login.	getProfile(token)
		Mengirim permintaan untuk memperbarui data profil (nama, no. hp, dan email) penjual yang sedang login.	updateProfile({ nama, no_hp, email, token })
27	PesananController	Mengirim permintaan untuk mengambil daftar pesanan yang sedang aktif (masuk) dengan sistem paginasi.	getPesananMasuk(token, page)
		Mengirim permintaan untuk mengambil detail lengkap dari satu pesanan masuk yang spesifik.	getDetailPesananMasuk(id, token)
		Mengirim permintaan untuk mendapatkan jumlah total pesanan yang sedang aktif, biasanya untuk notifikasi.	getCountPesananMasuk(token)
		Mengirim permintaan untuk memperbarui status sebuah pesanan.	updateStatusPesanan(id, newStatusKey, token)
		Mengirim permintaan untuk mengambil daftar pesanan yang sudah selesai (riwayat) dengan sistem paginasi.	getRiwayatPesanan(token, page)
		Mengirim permintaan untuk mengambil detail lengkap dari satu pesanan dalam riwayat.	getDetailRiwayatPesanan(id, token)
		Mengirim permintaan untuk membuat pesanan baru di database berdasarkan data dari keranjang belanja.	buatPesanan(payload)
		Mengirim permintaan untuk mengambil detail lengkap dari sebuah pesanan, biasanya untuk ditampilkan di halaman status pesanan.	getPesananDetail(pesanan_id, guest_id)
		Mengirim permintaan untuk hanya mengambil status terkini dari sebuah pesanan, digunakan untuk pengecekan berkala	getStatusPesananGuest(pesanan_id, guest_id)
28	AuthController	Mengirim permintaan untuk mendaftarkan akun penjual baru.	register({ nama, no_hp, email, password, confirmPassword })
		Mengirim permintaan untuk mendaftarkan kios baru yang terhubung dengan akun penjual.	registerKios({ nama_kios, nama_bank, nomor_rekening, token })
		Mengirim permintaan untuk memverifikasi kode OTP yang dikirim ke nomor WhatsApp penjual.	verifyOtp({ no_hp, kode_otp })
		Mengirim permintaan untuk mengirim ulang kode OTP ke nomor WhatsApp penjual.	resendOtp(no_hp)
		Mengirim permintaan untuk melakukan proses login penjual.	login(nama, password, rememberMe)
		Mengirim permintaan untuk memulai proses lupa kata sandi, dengan mengirim tautan reset ke WhatsApp.	forgotPassword(no_hp)
		Mengirim permintaan untuk mengatur ulang kata sandi penjual dengan yang baru, menggunakan token dari tautan reset.	resetPassword({ token, password, confirmPassword })
		Mengirim permintaan untuk melakukan proses logout, yang akan membuat token sesi tidak valid lagi.	logout(token)
29	KeranjangController	Mengirim permintaan untuk mengambil semua isi keranjang belanja milik pembeli.	getKeranjang(guest_id)
		Mengirim permintaan untuk menambahkan item baru ke dalam keranjang.	addItem(guest_id, menu_id, jumlah, catatan)
		Mengirim permintaan untuk memperbarui jumlah atau catatan dari item yang sudah ada di keranjang.	updateItem(guest_id, id, jumlah, catatan)
		Mengirim permintaan untuk menghapus satu item dari keranjang.	removeItem(guest_id, id)
30	PaymentController	Mengirim permintaan ke server untuk membuat transaksi baru di Midtrans.	buatTransaksiMidtrans(payload)
31	DashboardController	Mengirim permintaan untuk mengambil data ringkasan dashboard, seperti total pesanan, total menu, dan total pendapatan.	getDashboardData(token)
32	Penjual		Id
			Nama
			no_hp
			Email

No	Nama Kelas	Kandidat Perilaku	Nama Perilaku
33	Kios		Password
			Otp
			Id
			penjual_id
			nama_kios
			nama_bank
			nomor_rekening
			Deskripsi
			gambar_kios
34	Menu		Id
			penjual_id
			kios_id
			nama_menu
			Deskripsi
			Harga
			foto_menu
			status_tersedia
			estimasi_menit
35	Pesanan		Id
			nama_pemesan
			total_harga
			Catatan
			quest_id
			tipe_pengantaran
			no_hp
			diantar_ke
			Status
			total_estimasi
			kios_id
36	Keranjang		Id
			quest_id
			menu_id
			Jumlah
			total_harga
			kios_id
37	Pesanan Detail		Id
			pesanan_id
			menu_id
			nama_menu
			Jumlah
			Harga
			Subtotal
			foto_menu

4.3.2.4 Perancangan Kelas

Pada bagian ini dilakukan perancangan kelas, yaitu penyusunan diagram kelas dari kelas yang sudah didefinisikan beserta atribut dan operasinya. Berikut adalah diagram kelas untuk sistem yang dikembangkan.

Tabel 4. 43 Diagram Perancangan Kelas



Gambar 4. 47 Diagram Perancangan Kelas

Deskripsi

Diagram kelas ini menggambarkan alur: Penjual memiliki satu kios. Setiap Kios menyediakan banyak Menu. Pembeli menambahkan Menu ke Keranjang, lalu saat checkout terbentuk Pesanan. Rincian item pesanan dicatat sebagai PesananDetail (nama menu, harga, jumlah, subtotal). Penjual dapat melihat pesanan masuk dan memperbarui statusnya.

4.3.3 Pemodelan Berbasis Perilaku

Model perilaku berfungsi untuk menggambarkan cara kerja perangkat lunak ketika menerima masukan atau kejadian dari lingkungan luar. Melalui model ini dapat terlihat aliran proses, interaksi antar komponen, serta reaksi sistem terhadap setiap peristiwa. Pada bagian ini akan dijelaskan beberapa subbab terkait pemodelan perilaku, meliputi identifikasi event dan penyusunan diagram sekuen.

Tabel 4. 44 Pemodelan Berbasis Perilaku

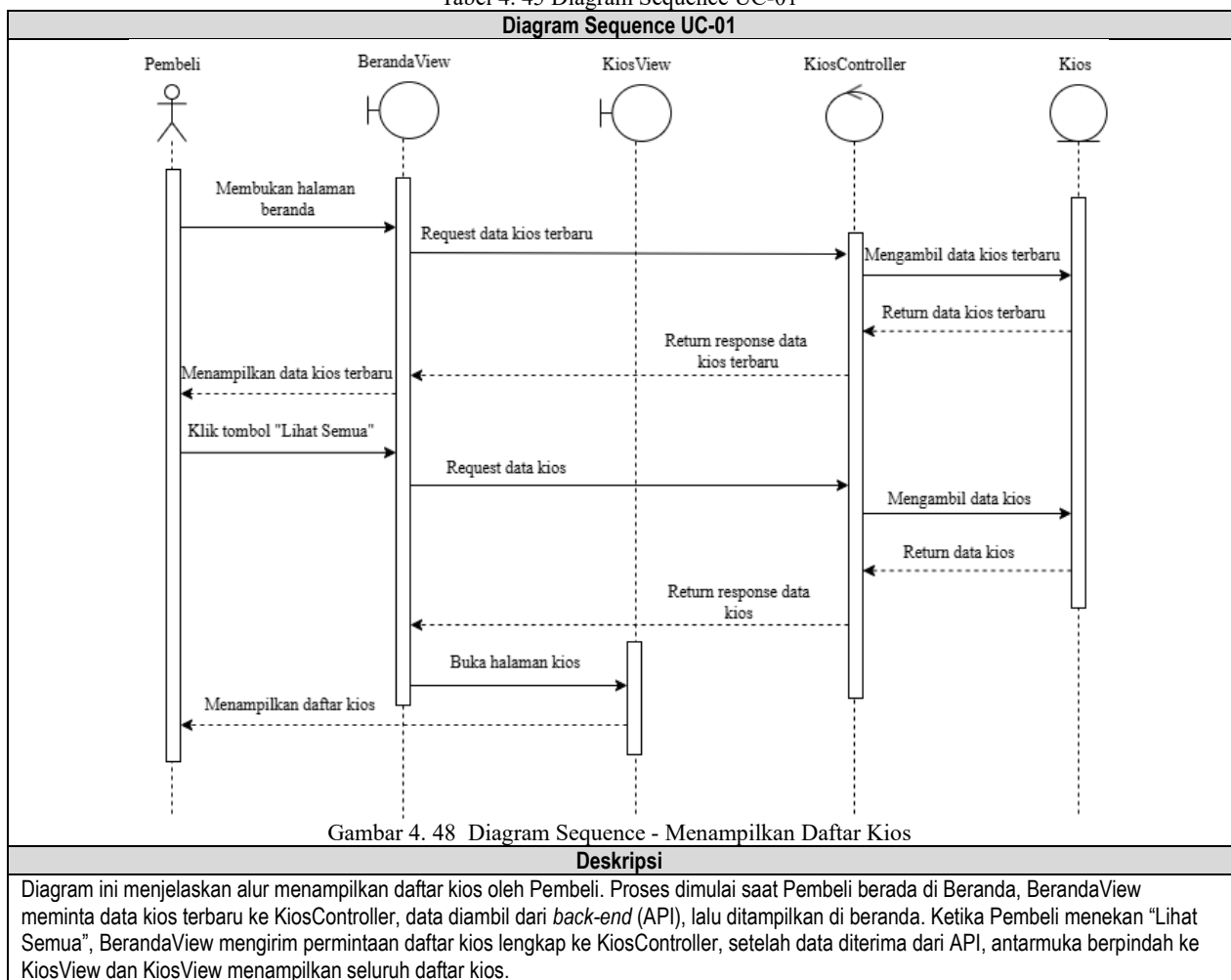
Kode	Nama Event	Aktor
UC-01	Menampilkan daftar kios	Pembeli
UC-02	Menampilkan daftar menu perkios	Pembeli
UC-03	Menampilkan detail menu	Pembeli
UC-04	Mencari menu	Pembeli
UC-05	Mengelola keranjang	Pembeli
UC-06	Membuat pesanan dari keranjang	Pembeli
UC-07	Menyediakan Payment gateway	Pembeli
UC-08	Menampilkan status dan estimasi waktu pesanan	Pembeli
UC-09	Mengirim notifikasi pesanan selesai	Pembeli
UC-10	Mengelola Menu	Penjual
UC-11	Mengirim notifikasi pesanan masuk	Penjual
UC-12	Menampilkan daftar pesanan dan detail pesanan	Penjual
UC-13	Mengubah status pesanan	Penjual

Kode	Nama Event	Aktor
UC-14	Menampilkan Tipe Pengantaran	Penjual

4.3.3.1 Diagram Sequence

Diagram Sequence pada dasarnya menggambarkan bagaimana sistem akan berperilaku dalam menanggapi tindakan atau event yang datang dari luar dengan menampilkan urutan interaksi antar aktor dan komponen sistem. Diagram ini memperlihatkan siapa yang terlibat, pesan apa yang dikirim, serta bagaimana status atau data berubah seiring alur waktu, sehingga memudahkan dalam memahami respon sistem terhadap skenario yang dijalankan.

Tabel 4. 45 Diagram Sequence UC-01



Tabel 4. 46 Diagram Sequence UC-02

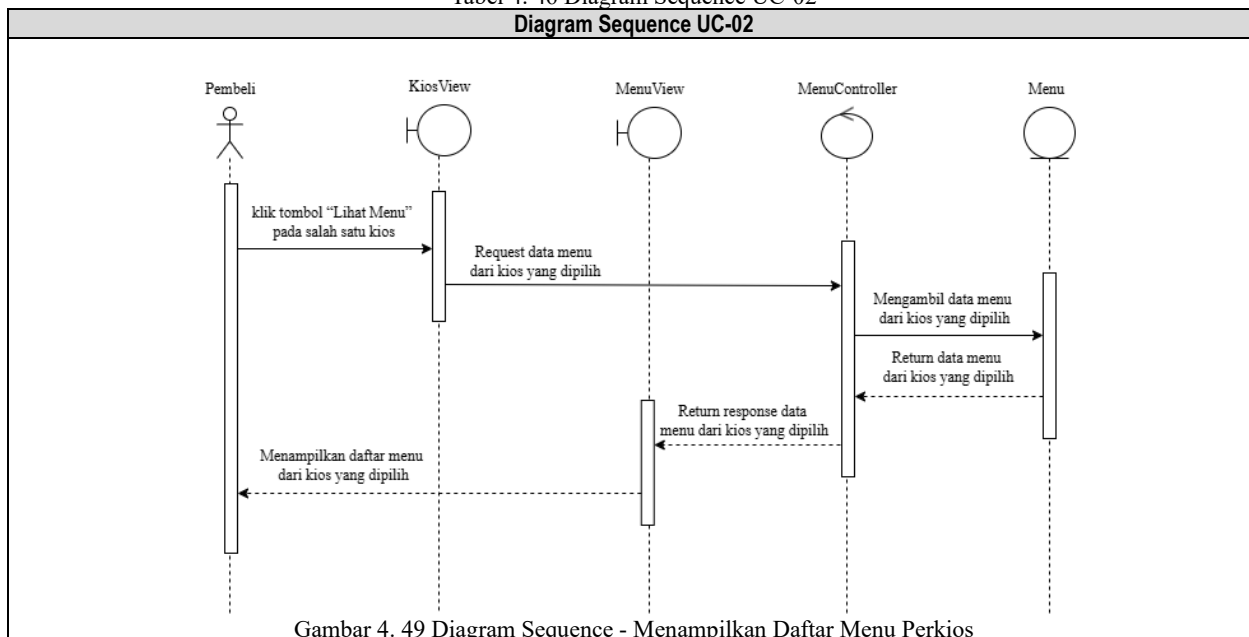
**Deskripsi**

Diagram ini menjelaskan alur menampilkan daftar menu per kios oleh Pembeli. Proses dimulai saat Pembeli menekan tombol "Lihat Menu" pada salah satu kios di KiosView, antarmuka mengirim permintaan data menu kios terpilih ke MenuController. Controller mengambil data dari *back-end* (API) dan mengembalikan respons. Setelah data diterima, antarmuka membuka MenuView dan menampilkan daftar menu milik kios yang dipilih kepada Pembeli.

Tabel 4. 47 Diagram Sequence UC-03

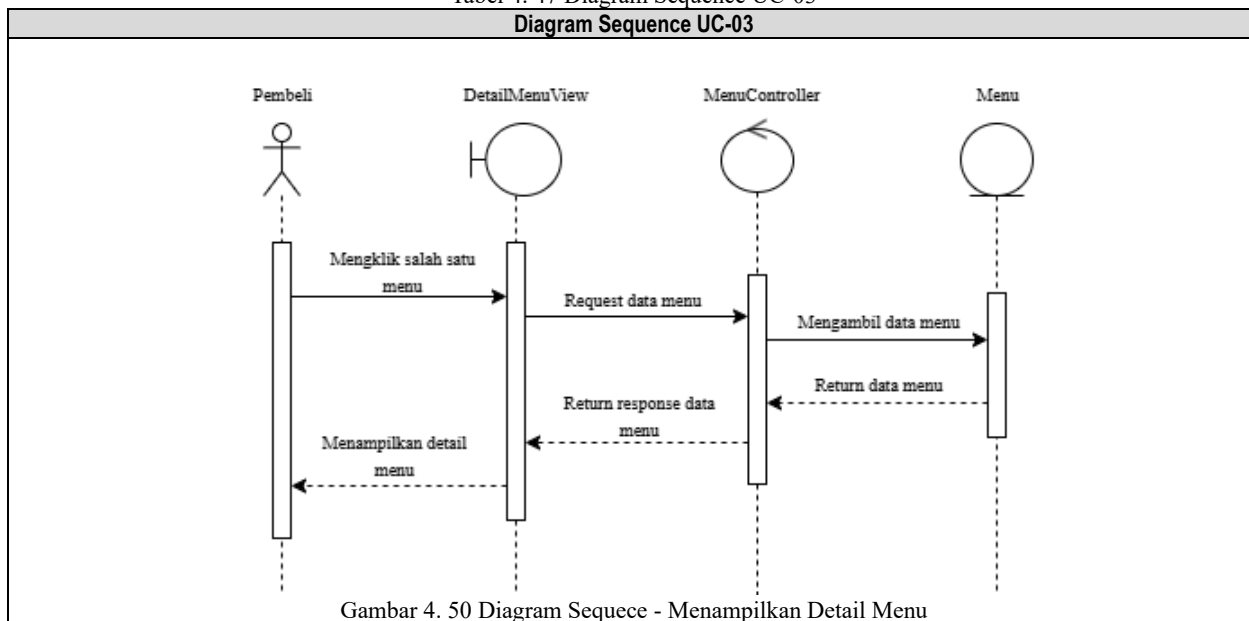
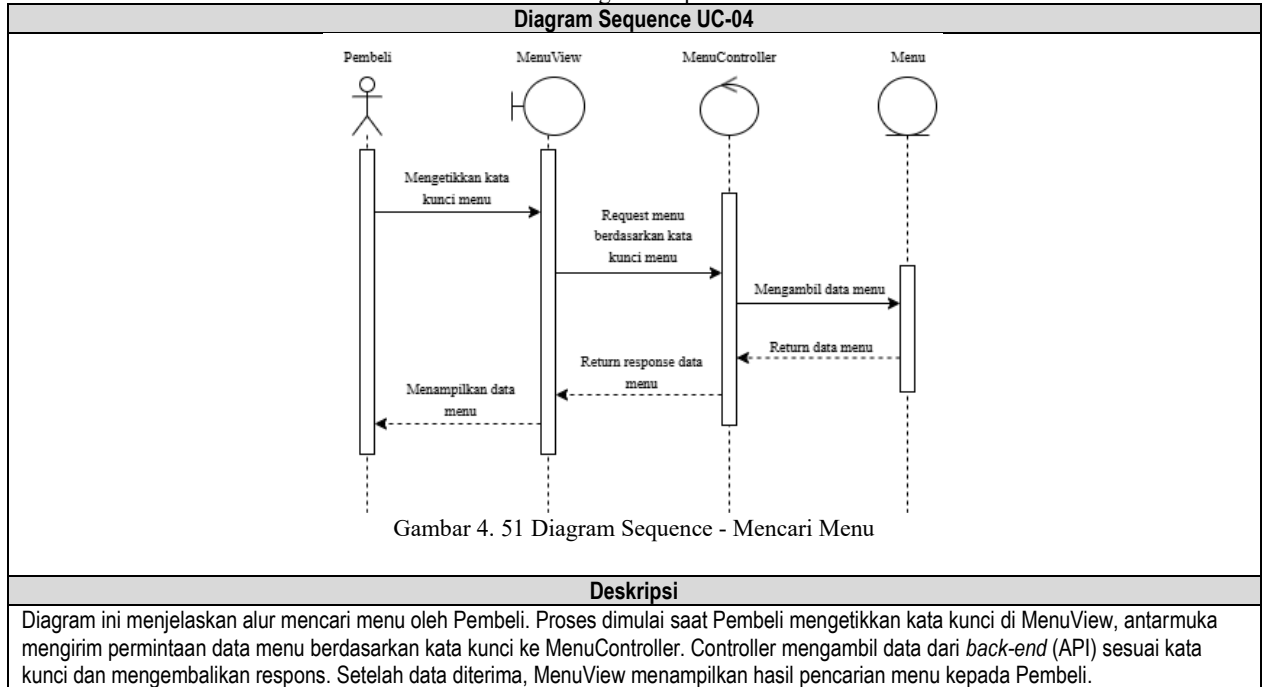
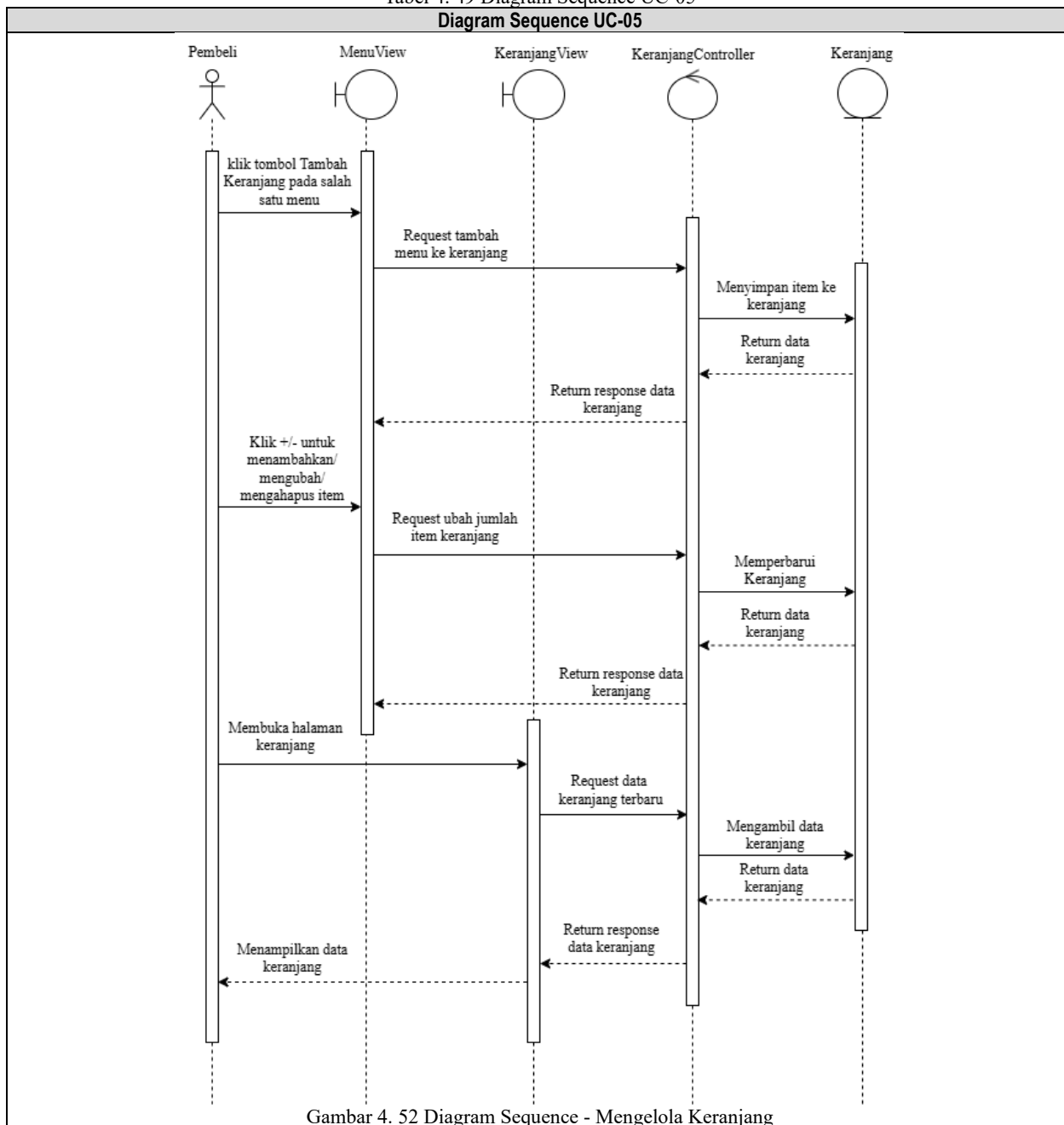
**Deskripsi**

Diagram ini menjelaskan alur menampilkan detail menu oleh Pembeli. Proses dimulai saat Pembeli mengklik salah satu menu; DetailMenuView mengirim permintaan data menu ke MenuController. Controller mengambil data dari *back-end* (API) lalu mengembalikan respons ke antarmuka. Setelah data diterima, DetailMenuView menampilkan detail menu yang dipilih kepada Pembeli.

Tabel 4. 48 Diagram Sequence UC-04



Tabel 4. 49 Diagram Sequence UC-05

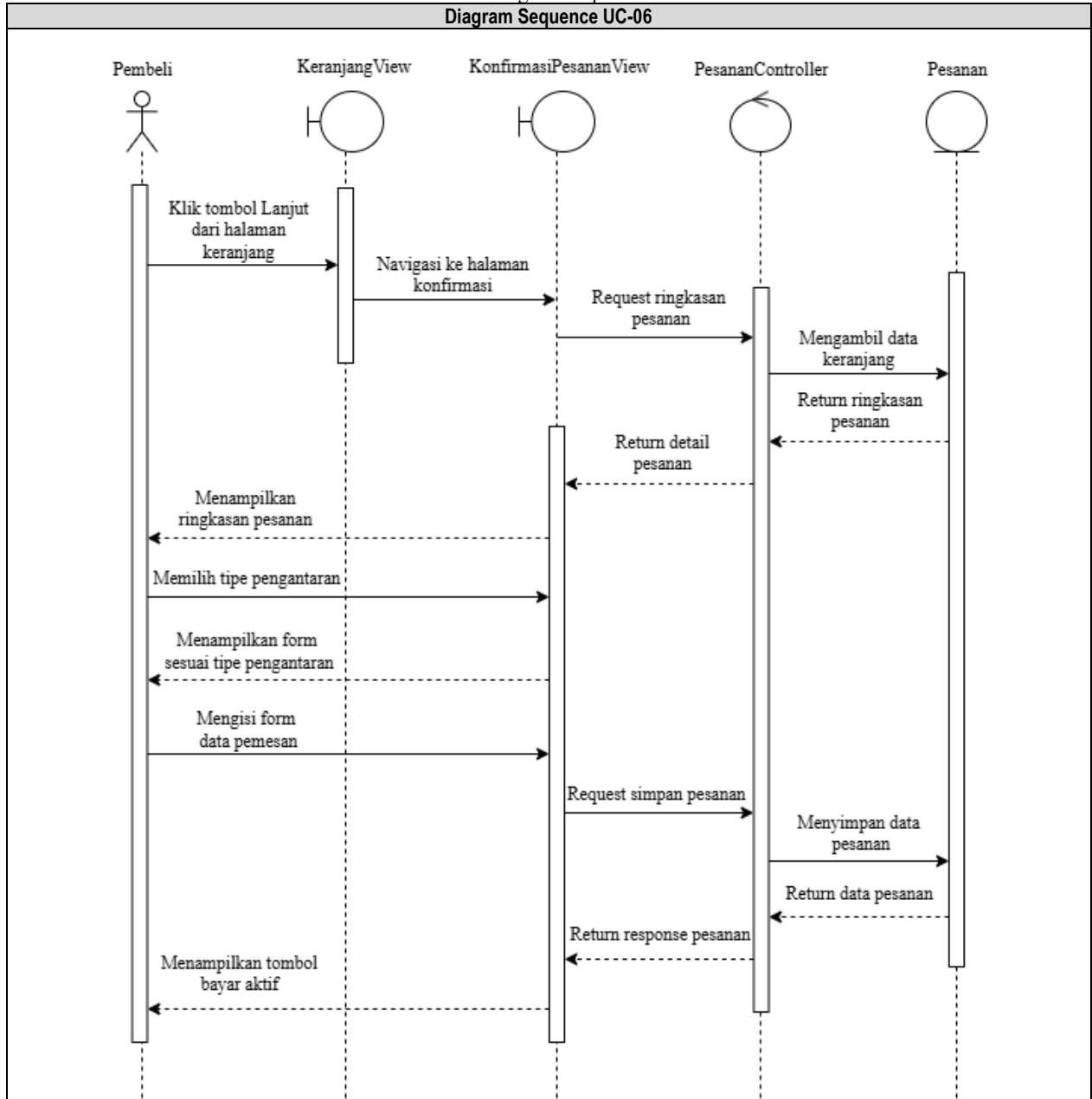


Gambar 4. 52 Diagram Sequence - Mengelola Keranjang

Deskripsi

Diagram ini menggambarkan alur mengelola keranjang oleh Pembeli: saat menekan Tambah Keranjang pada card di MenuView, permintaan dikirim ke KeranjangController untuk menyimpan item ke API Keranjang dan mengirim data keranjang terbaru, setelah itu Pembeli dapat menekan +/- untuk mengubah jumlah, MenuView mengirim permintaan ubah jumlah, controller memperbarui data ke API, item otomatis dihapus ketika jumlah menjadi 0, lalu mengirim data keranjang terbaru, ketika Pembeli membuka halaman Keranjang, KeranjangView meminta data keranjang terbaru ke controller, controller mengambil dari API dan mengembalikannya sehingga daftar pesanan terbaru ditampilkan.

Tabel 4. 50 Diagram Sequence UC-06

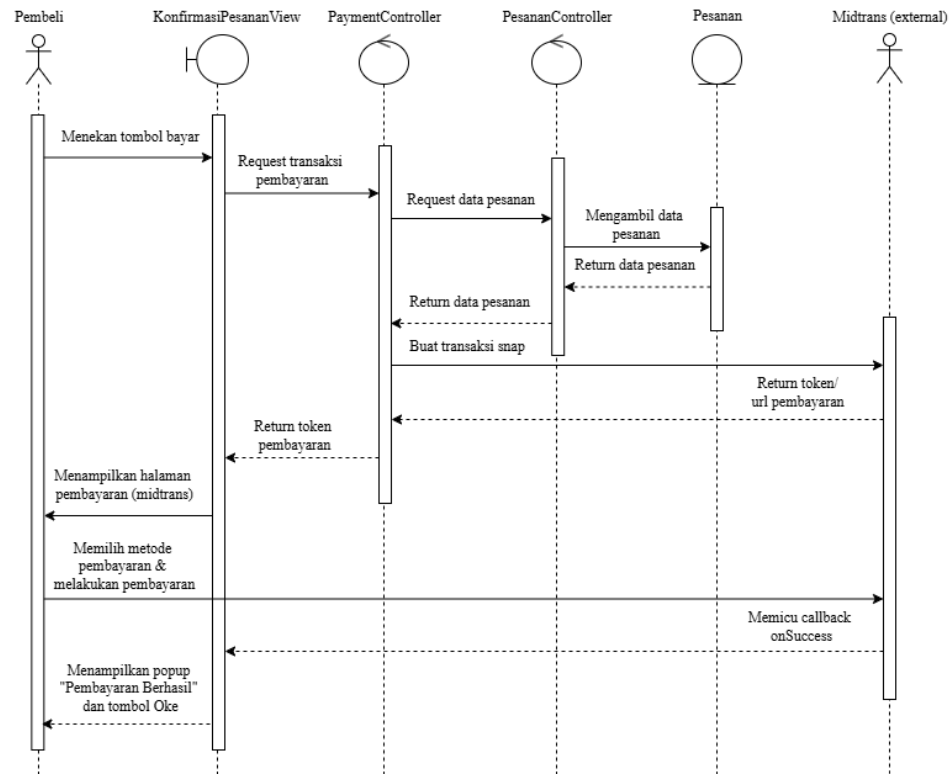


Gambar 4. 53 Diagram Sequence - Membuat Pesanan dari Keranjang

Deskripsi

Diagram ini menjelaskan alur membuat pesanan dari keranjang. Setelah Pembeli menekan Lanjut di KeranjangView, aplikasi menavigasi ke KonfirmasiPesananView yang kemudian request ringkasan pesanan ke PesananController; controller mengambil data keranjang dari API dan return detail pesanan untuk ditampilkan sebagai ringkasan. Pembeli memilih tipe pengantaran, tampilan menyesuaikan form sesuai pilihan, lalu Pembeli mengisi data pemesan (Nama, No WhatsApp, Catatan, alamat bila Diantar). Selanjutnya KonfirmasiPesananView request simpan pesanan ke controller, controller menyimpan pesanan ke API dan return response pesanan, sehingga tombol Bayar menjadi aktif.

Tabel 4. 51 Diagram Sequence UC-07

Diagram Sequence UC-07

Gambar 4. 54 Diagram Sequence - Menyediakan Payment Gateway

Deskripsi

Diagram ini menjelaskan alur proses pembayaran. Setelah Pembeli menekan tombol Bayar di KonfirmasiPesananView, tampilan mengirim request transaksi pembayaran ke PaymentController. PaymentController request data pesanan ke PesananController, PesananController mengambil data pesanan dari API dan return data pesanan. PaymentController kemudian membuat transaksi Snap ke Midtrans; Midtrans return token/URL pembayaran. Token tersebut dikembalikan ke KonfirmasiPesananView untuk menampilkan halaman pembayaran Midtrans. Pembeli memilih metode dan menyelesaikan pembayaran; Midtrans memicu callback onSuccess ke *back-end*. Aplikasi lalu menampilkan popup "Pembayaran Berhasil" beserta tombol Oke.

Tabel 4. 52 Diagram Sequence UC-08

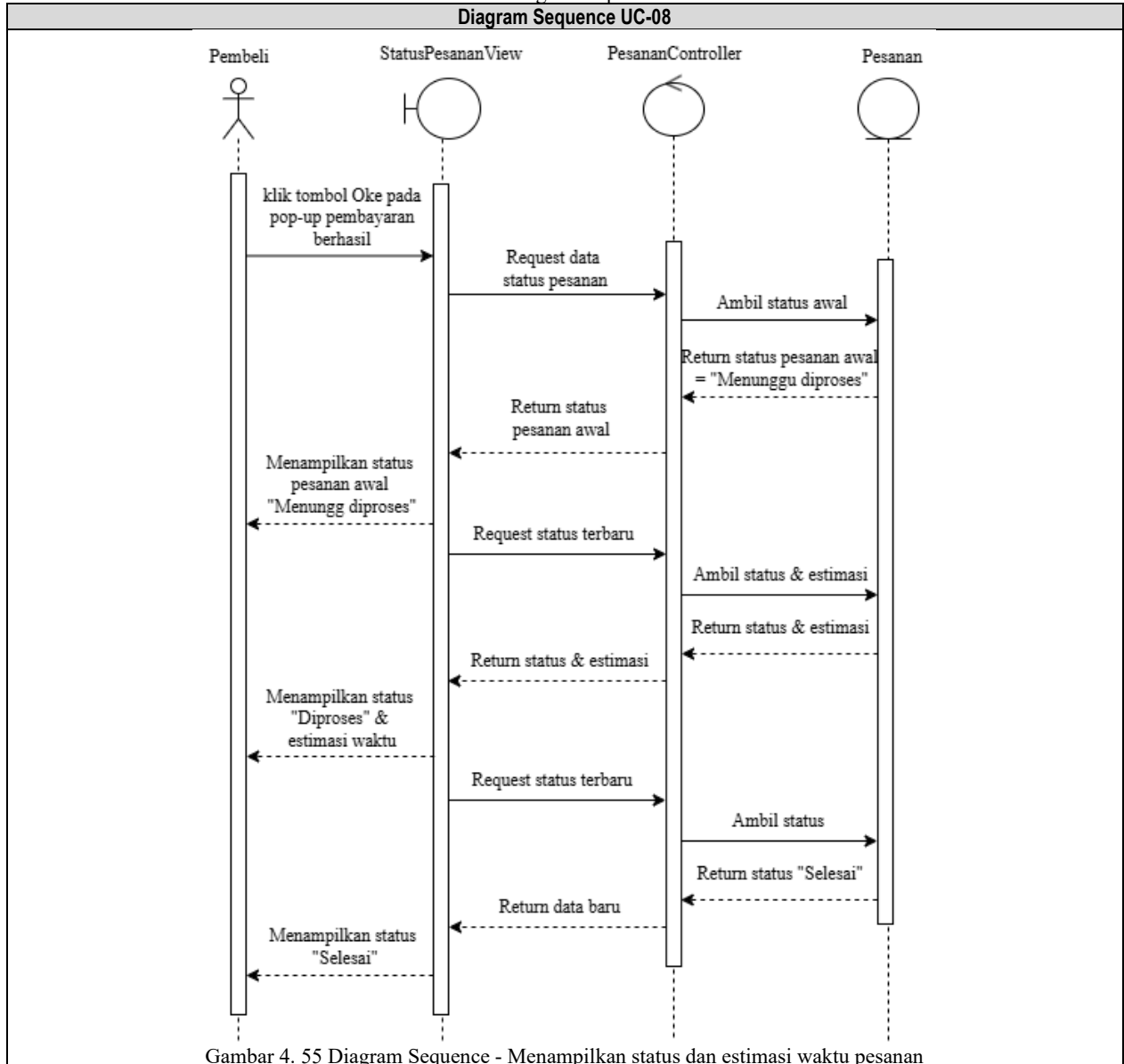
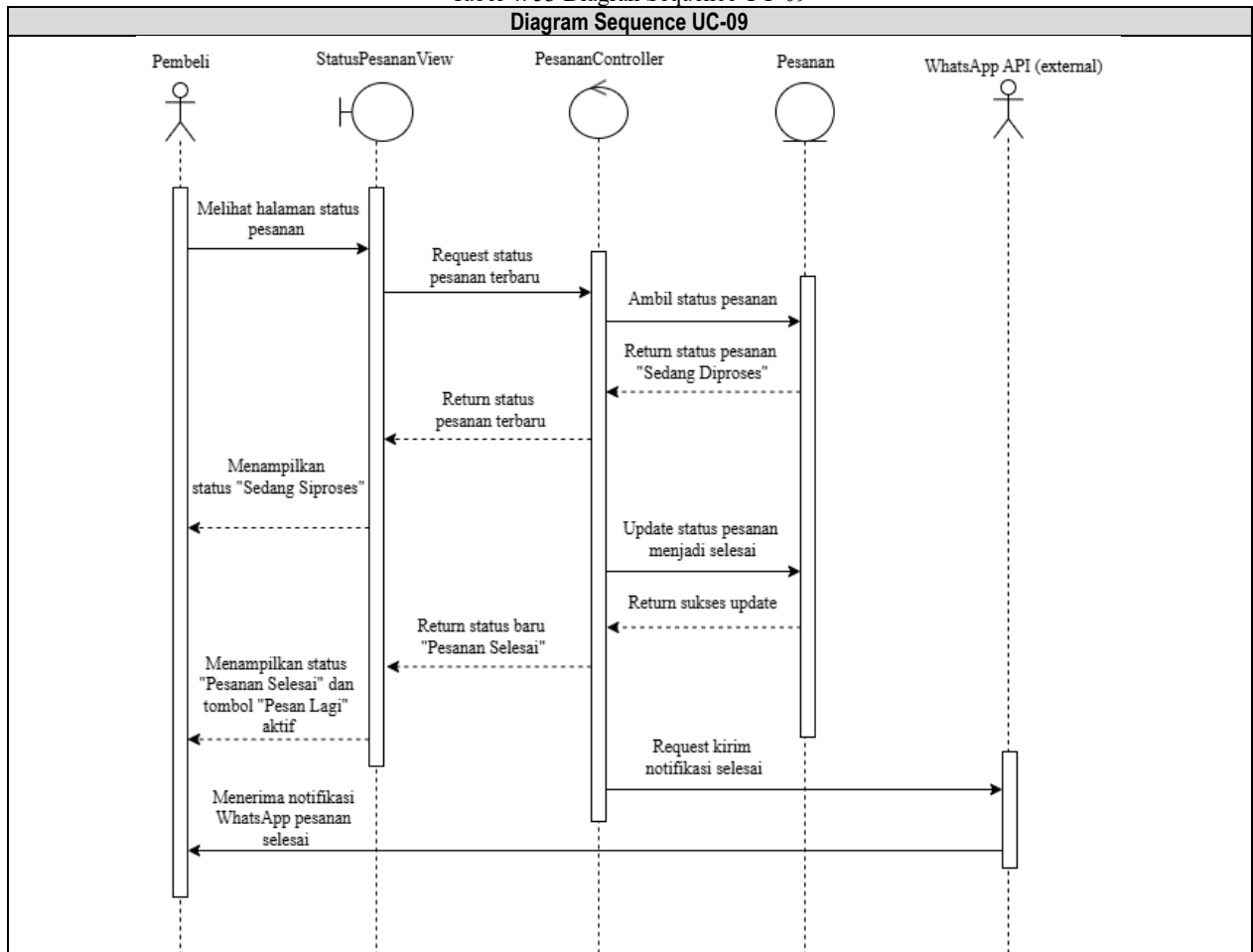
**Deskripsi**

Diagram ini menjelaskan alur menampilkan status dan estimasi pesanan. Setelah Pembeli menekan Oke pada pop-up pembayaran berhasil, aplikasi membuka StatusPesananView yang request status awal ke PesananController, controller mengambil status dari API Pesanan dan return status awal "Menunggu diproses". View lalu request status terbaru secara berkala, controller mengembalikan status dan estimasi waktu, sehingga ditampilkan "Diproses" beserta estimasinya. Ketika API mengembalikan status akhir, controller return status "Selesai" dan View menampilkan "Selesai".

Tabel 4. 53 Diagram Sequence UC-09

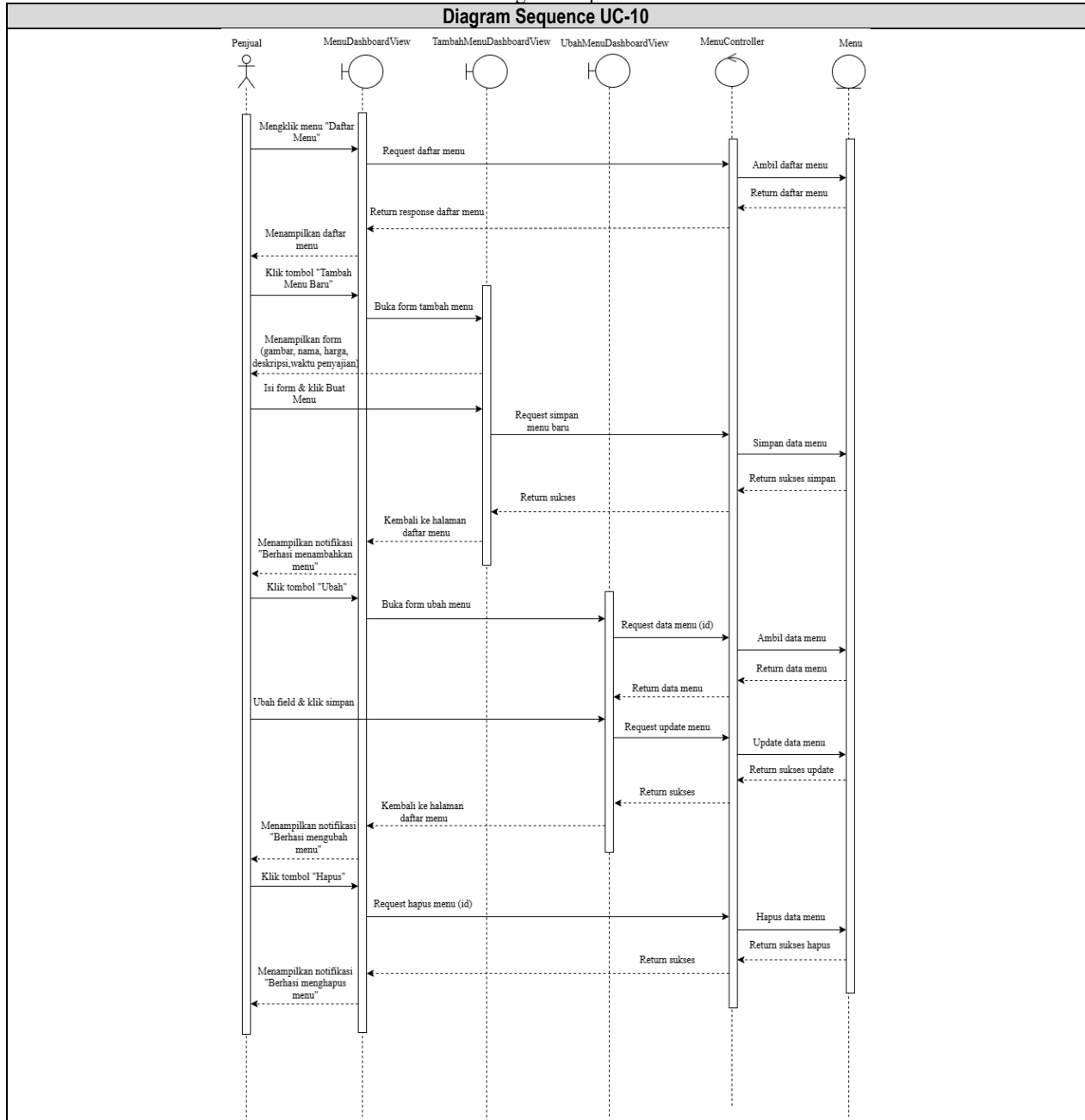


Gambar 4. 56 Diagram Sequence - Menerima Notifikasi Pesanan Selesai

Deskripsi

Diagram ini menjelaskan alur pembeli menerima notifikasi pesanan selesai. Pembeli melihat halaman status pesanan, lalu StatusPesananView meminta status terbaru ke PesananController. Controller mengambil data dari Pesanan dan mengembalikan status "Sedang Diproses" untuk ditampilkan. Ketika pesanan selesai, controller memperbarui status pada Pesanan dan menerima konfirmasi sukses, kemudian mengirim status baru "Pesanan Selesai" ke View sehingga ditampilkan kepada pembeli bersama tombol Pesan Lagi aktif. Terakhir, controller juga mengirim request ke WhatsApp API, dan pembeli menerima notifikasi WhatsApp bahwa pesanan sudah selesai.

Tabel 4. 54 Diagram Sequence UC-10

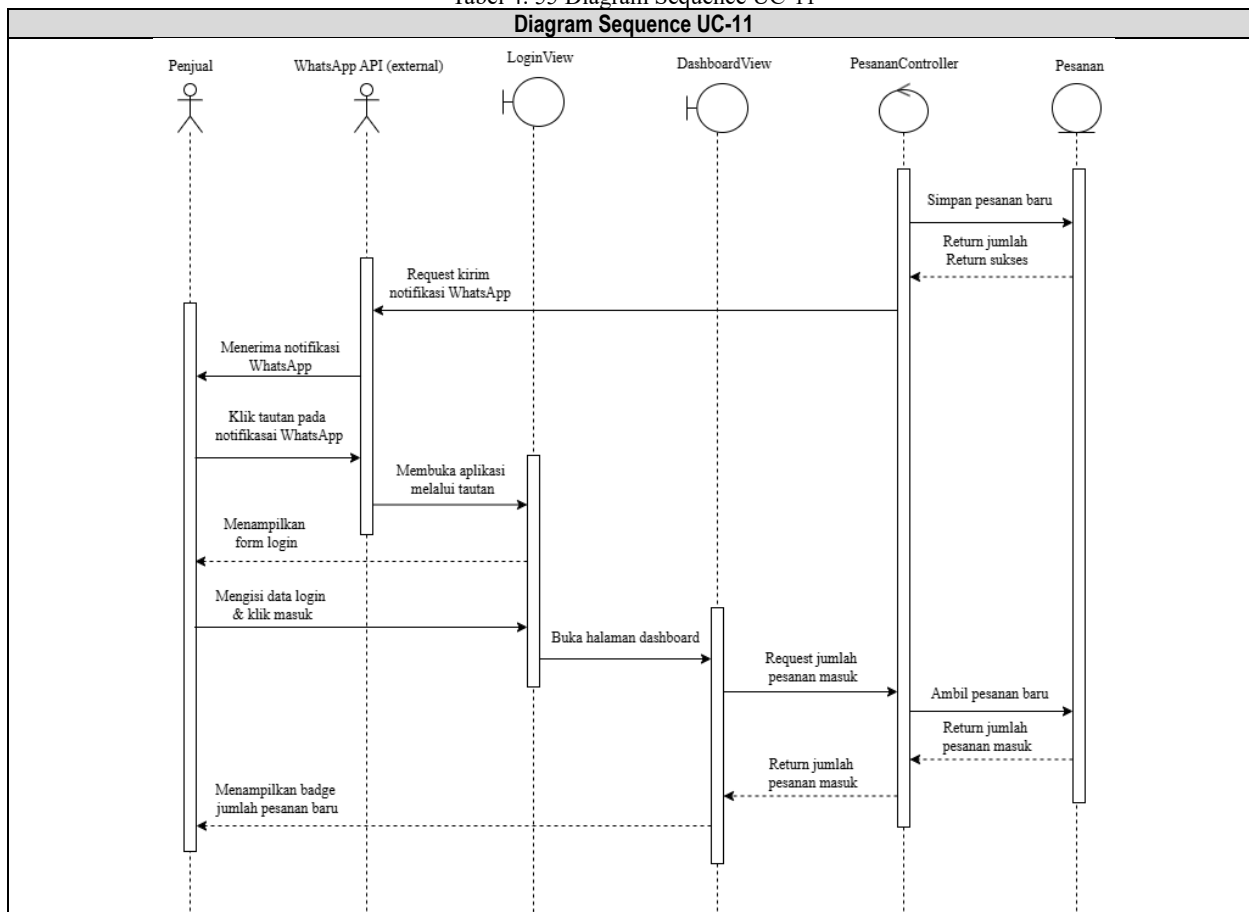


Gambar 4. 57 Diagram Sequence - Mengelola Menu

Deskripsi

Diagram ini menjelaskan alur mengelola menu oleh Penjual. Saat Penjual membuka MenuDashboardView, tampilan request daftar menu ke MenuController, controller mengambil data dari API Menu dan return daftar menu untuk ditampilkan. Penjual menekan "Tambah Menu Baru", form terbuka, Penjual isi form & klik Buat Menu, request simpan menu baru, controller menyimpan ke API dan return sukses, lalu kembali ke daftar dengan notifikasi berhasil menambahkan. Untuk ubah, Penjual menekan "Ubah", form ubah terbuka, tampilan request data menu (id), controller ambil dari API dan return data, Penjual ubah field & klik Simpan, request update menu, controller update ke API dan return sukses, daftar diperbarui dengan notifikasi berhasil mengubah. Untuk hapus, Penjual menekan "Hapus", request hapus menu (id), controller hapus di API dan return sukses, daftar diperbarui dengan notifikasi berhasil menghapus.

Tabel 4. 55 Diagram Sequence UC-11

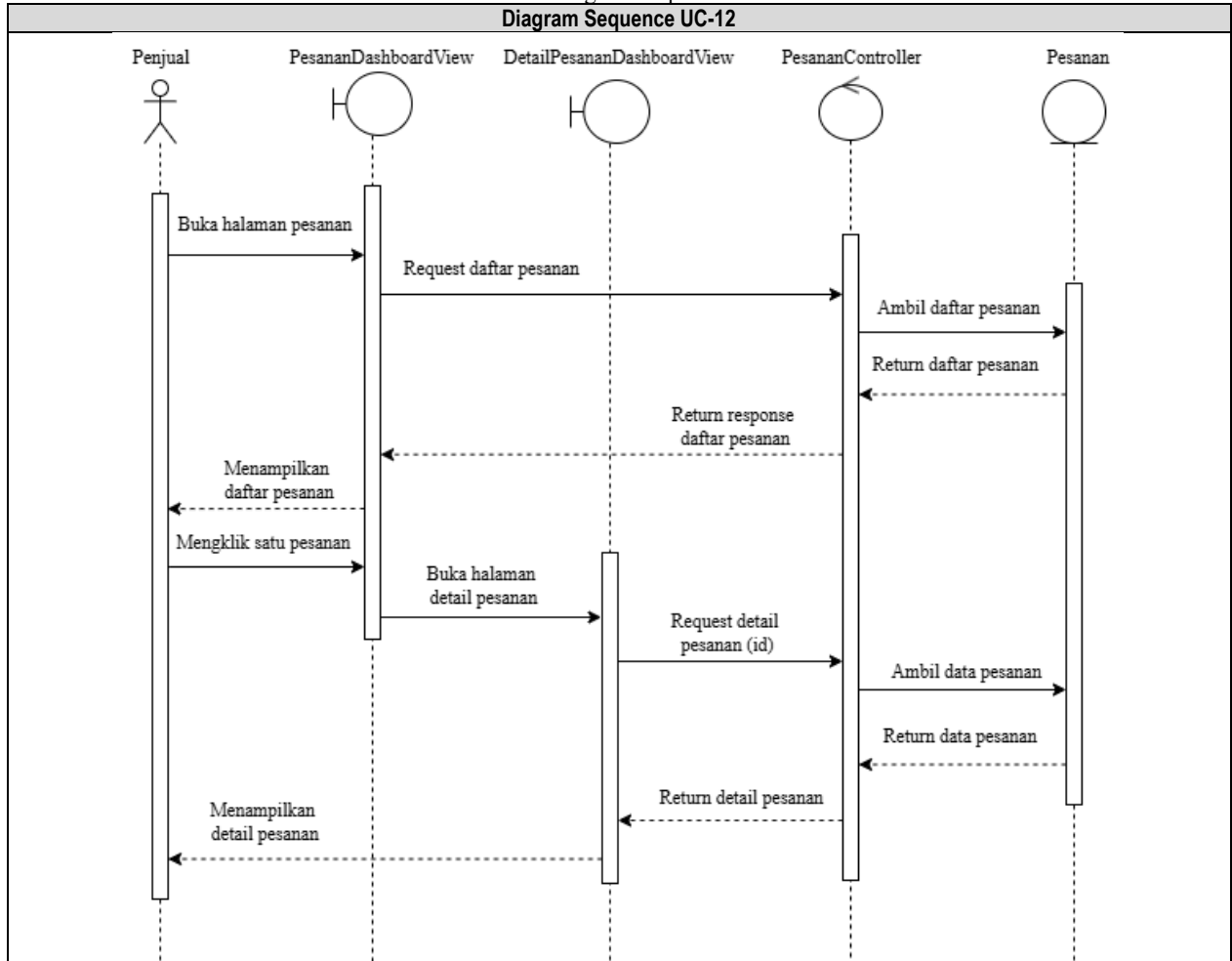


Gambar 4. 58 Diagram Sequence - Menerima Notifikasi Pesanan Masuk

Deskripsi

Diagram ini menjelaskan alur penjual menerima notifikasi pesanan baru melalui WhatsApp. Setelah ada pesanan baru tersimpan di Pesanan (DB), PesananController mengirim request ke WhatsApp API untuk meneruskan notifikasi. WhatsApp API kemudian mengirim pesan WhatsApp ke Penjual berisi tautan menuju aplikasi. Penjual menekan tautan tersebut sehingga aplikasi terbuka pada LoginView. Jika sesi belum aktif, sistem menampilkan form login, penjual mengisi data login dan masuk kembali. Setelah itu, aplikasi membuka DashboardView. Dari Dashboard, sistem mengirim request ke PesananController untuk mendapatkan jumlah pesanan baru. Controller mengambil data dari Pesanan (DB) dan mengembalikan jumlah pesanan masuk ke Dashboard. Akhirnya, Dashboard menampilkan badge jumlah pesanan baru kepada penjual.

Tabel 4. 56 Diagram Sequence UC-12

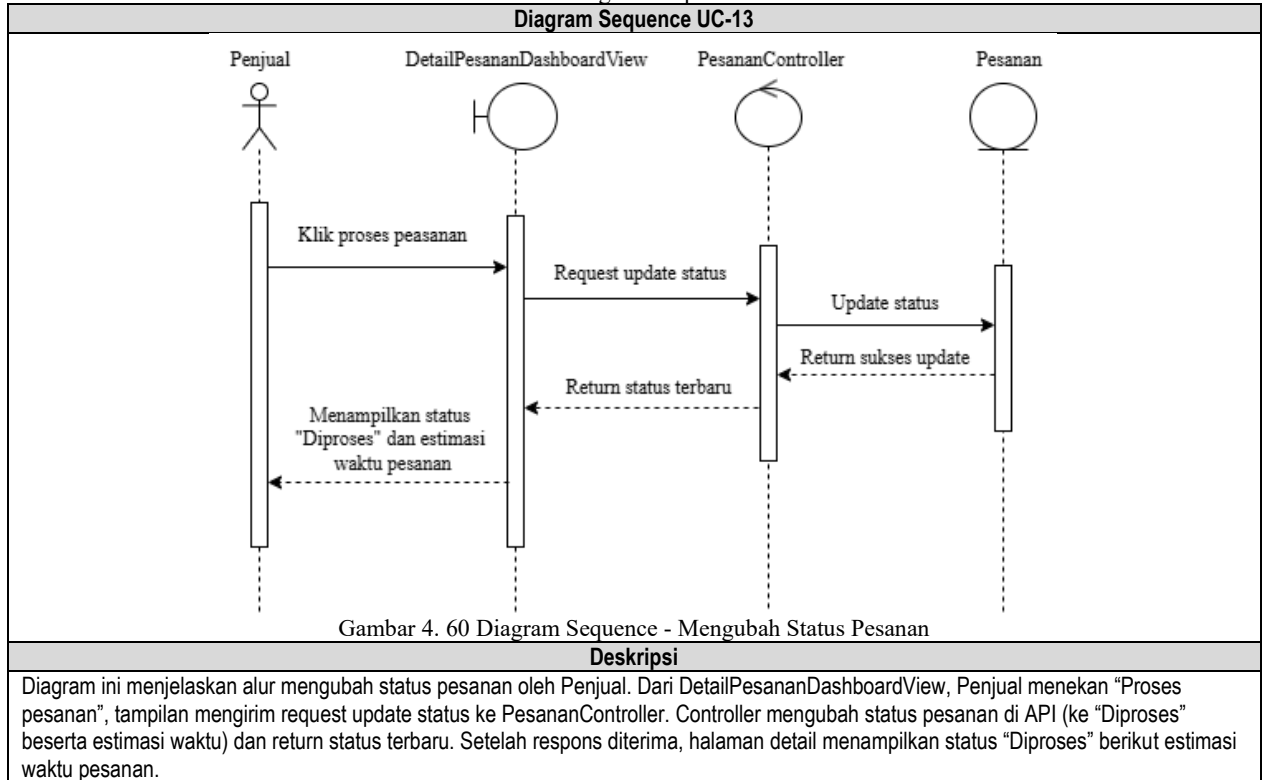


Gambar 4. 59 Diagram Sequence - Menampilkan Daftar Pesanan dan Detail Pesanan

Deskripsi

Diagram ini menjelaskan alur menampilkan daftar pesanan dan detail pesanan oleh Penjual. Saat Penjual membuka halaman pesanan, PesananDashboardView melakukan request daftar pesanan ke PesananController, controller mengambil daftar dari API Pesanan dan return daftar pesanan untuk ditampilkan. Ketika Penjual mengklik salah satu pesanan, tampilan menavigasi ke DetailPesananDashboardView lalu request detail pesanan (id) ke controller, controller mengambil data detail dari API dan return detail pesanan, sehingga halaman detail menampilkan informasi lengkap pesanan terpilih.

Tabel 4. 57 Diagram Sequence UC-13



Tabel 4. 58 Diagram Sequence UC-14

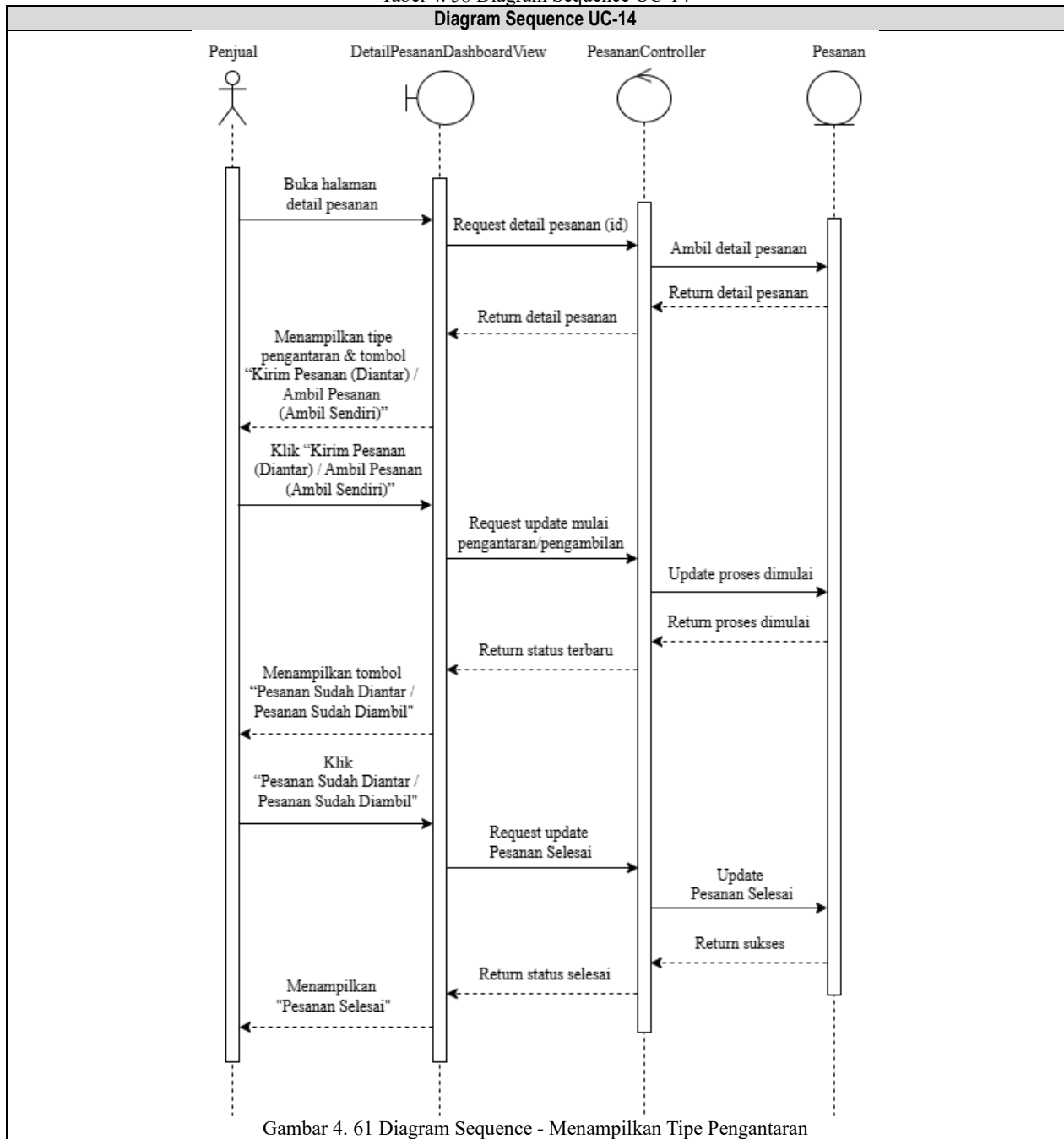
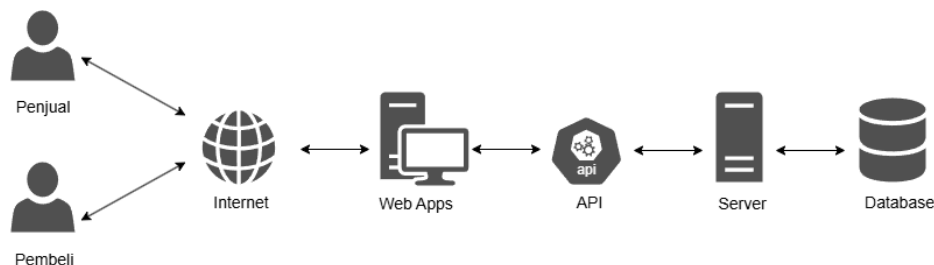
**Deskripsi**

Diagram ini menjelaskan alur menampilkan tipe pengantaran oleh Penjual. Dari DetailPesananDashboardView, sistem memuat detail pesanan (termasuk tipe pengantaran) dari PesananController. Berdasarkan tipe, halaman menampilkan tombol utama "Kirim Pesanan (Diantar)" atau "Ambil Pesanan (Ambil Sendiri)". Saat tombol ditekan, view mengirim request update mulai pengantaran/pengambilan ke controller, controller memperbarui status di API dan return status terbaru. Setelah proses selesai, Penjual menekan "Pesanan Sudah Diantar/Diambil"; view mengirim request update selesai, controller menyimpan perubahan dan return "Pesanan Selesai", lalu halaman menampilkan status "Pesanan Selesai."

4.3.4 Perancangan Arsitektur Aplikasi

Perancangan arsitektur aplikasi digunakan untuk menggambarkan struktur serta komunikasi antar komponen sistem, seperti *front-end*, *back-end*, basis data, dan layanan pendukung. Rancangan ini menjadi acuan utama dalam membangun dan mengembangkan aplikasi. Berikut merupakan rancangan arsitektur yang digunakan pada penelitian tugas akhir.



Gambar 4. 62 Perancangan Arsitektur Aplikasi

Arsitektur sistem ini dibangun dengan konsep *client-server* di mana semua pengguna mengakses aplikasi melalui browser. Seluruh interaksi yang dilakukan oleh penjual maupun pembeli akan diteruskan menuju layanan *back-end* agar dapat diproses. Komponen utama dalam arsitektur ini meliputi:

1. Penjual dan Pembeli

- **Penjual** memanfaatkan aplikasi web untuk menambahkan atau memperbarui produk serta melihat pesanan yang masuk.
- **Pembeli** menggunakan aplikasi web untuk mencari produk, melakukan pemesanan, dan melanjutkan ke proses pembayaran.

2. Internet

Berfungsi sebagai jalur komunikasi utama yang menghubungkan pengguna dengan sistem; semua pertukaran data antara aplikasi web dan layanan di sisi *back-end* berlangsung melalui internet menggunakan protokol standar seperti HTTP.

3. Aplikasi Web (Web Apps)

Merupakan antarmuka berbasis web yang menjadi media interaksi antara penjual dan pembeli dengan *system*, setiap aktivitas pengguna diterjemahkan menjadi permintaan data yang kemudian dikirim ke API melalui internet.

4. API (Application Programming Interface)

API berperan sebagai pintu masuk ke sistem *back-end*. Semua permintaan dari aplikasi web diterima oleh API, kemudian diproses atau diteruskan ke server. Data yang sudah diolah akan dikembalikan melalui API untuk ditampilkan di sisi aplikasi web.

5. Server

Server bertanggung jawab terhadap logika bisnis dan mengelola jalannya aplikasi di sisi *back-end*.

Server juga mengatur pertukaran data dengan database agar semua permintaan dari pengguna dapat dilayani.

6. Database

Berfungsi sebagai penyimpanan utama yang menyimpan data penjual, menu, pesanan, hingga riwayat pesanan. Database memungkinkan server untuk melakukan operasi pengelolaan data sesuai kebutuhan aplikasi.

BAB 5

SCRUM PROCESS DAN DELIVERY

Bab ini menjelaskan proses implementasi dan pengembangan *front-end* sistem pemesanan makanan online di Kantin Universitas Pasundan Setiabudi. Pengembangan menggunakan metode Scrum, yang menekankan iterasi dan kolaborasi. Pembahasan mencakup tahapan utama Scrum, mulai dari penyusunan Product Backlog, perencanaan dan pelaksanaan Sprint untuk membangun fitur inti, hingga tahap Delivery berupa produk fungsional.

5.1 Scrum Process

Pada tahap ini dibahas proses pengembangan perangkat lunak menggunakan metode Scrum. Sebelum menjelaskan tiap aktivitas Scrum secara mendetail, bagian ini akan memaparkan terlebih dahulu lingkungan teknis yang digunakan, seperti perangkat lunak pendukung, struktur folder proyek, dan pengaturan konfigurasi aplikasi. Setelah itu, pembahasan dilanjutkan dengan rangkaian aktivitas Scrum yang dilakukan secara iteratif selama pengembangan produk, yang akan dijelaskan lebih rinci pada subbab berikutnya.

1. Kakas Perangkat Lunak

Kakas perangkat lunak adalah alat bantu yang digunakan dalam proses pengembangan. Pada penelitian ini, kakas perangkat lunak dipakai untuk mendukung pembangunan *front-end* web pemesanan makanan online di Kantin Universitas Pasundan Setiabudi, berikut adalah kakas yang akan digunakan.

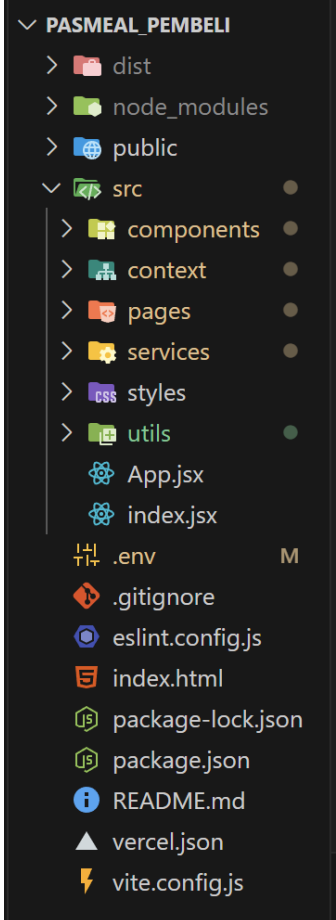
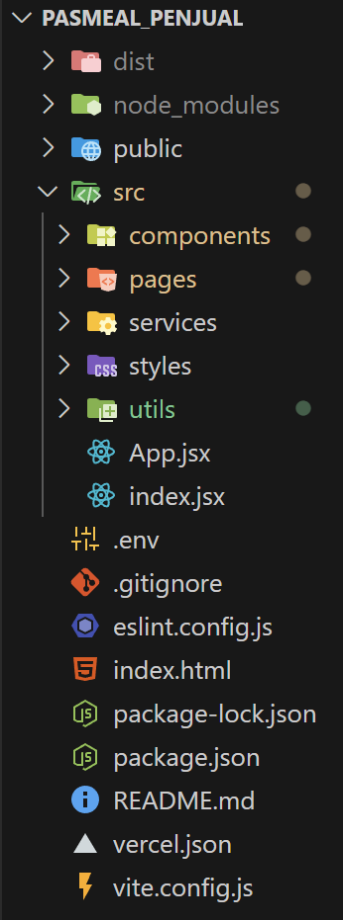
Tabel 5. 1 Kakas Perangkat Lunak

No	Nama	Deskripsi
1	React JS	<i>Library</i> JavaScript untuk membangun antarmuka pengguna (UI) yang interaktif dan berbasis komponen.
2	Tailwind CSS	<i>Framework</i> CSS yang menyediakan <i>utility class</i> sehingga memudahkan styling tampilan secara cepat dan konsisten.
3	NPM (Node Package Manager)	Manajer paket untuk mengelola dependensi dan <i>library</i> JavaScript yang dibutuhkan dalam proyek.
4	Visual Studio Code	Teks <i>editor</i> yang digunakan untuk menulis, mengelola, dan mengembangkan kode program.
5	Git	Sistem kontrol versi untuk mencatat perubahan kode.
6	GitHub	Platform penyimpanan repositori berbasis Git yang digunakan untuk mengelola proyek secara online.
7	Vercel	Vercel adalah <i>platform cloud hosting</i> modern yang dirancang khusus untuk mengembangkan dan meng- <i>deploy</i> aplikasi web berbasis <i>front-end</i> .

2. Struktur Folder

Berikut adalah struktur direktori utama yang dibuat dan disesuaikan sepanjang sprint ini:

Tabel 5. 2 Struktur Folder

Struktur Folder	
	

Gambar 5. 1 Struktur Folder

Deskripsi

Struktur direktori ini menggambarkan arsitektur dari aplikasi *front-end* PasMeal. Pada proyek React modern (dengan Vite), struktur dapat dibagi menjadi beberapa bagian utama:

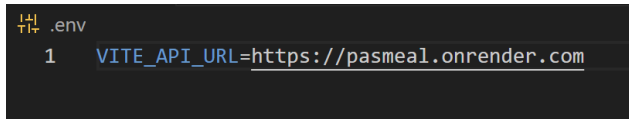
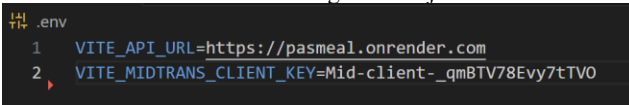
1. **src/** : Folder utama untuk seluruh kode aplikasi React.
 - **pages/** : Berisi halaman penuh seperti halaman Beranda, Menu, Kios. Setiap file mewakili satu rute aplikasi.
 - **components/** : Kumpulan komponen kecil yang bisa dipakai ulang, misalnya Button, atau Card Menu.
 - **context/** : Tempat menyimpan state global menggunakan React Context, seperti CartContext untuk keranjang.
 - **services/** : Menyimpan fungsi untuk komunikasi dengan API *back-end*, misalnya mengambil data menu atau membuat pesanan.
 - **styles/** : Berisi file CSS atau styling global untuk menjaga tampilan aplikasi tetap konsisten.
 - **Utils/** : Berisi helper function yang mendukung aplikasi. Saat ini terdapat imageHelper.js dengan fungsi getImageUrl(), yang digunakan untuk memastikan semua gambar memiliki URL valid.
 - **App.jsx** : Komponen utama aplikasi yang mengatur routing dan struktur halaman.
 - **index.jsx** : Entry point aplikasi, merender <App /> ke dalam index.html.
2. **dist/** : Hasil build aplikasi setelah menjalankan npm run build. Isinya sudah dioptimasi dan siap untuk deployment ke server/hosting.
3. **node_modules/** : Folder otomatis hasil dari npm install yang berisi semua dependency (React, Axios, Tailwind, dll). Folder ini tidak perlu diubah manual.
4. **public/** : Folder untuk file statis seperti logo, atau gambar yang bisa diakses langsung melalui URL.

5. **.env** : File berisi variabel lingkungan, seperti VITE_API_URL. Memudahkan pengaturan konfigurasi tanpa harus menulis langsung di kode.
6. **.gitignore** : File yang menentukan folder atau file apa saja yang tidak akan dikirim ke repository Git, misalnya node_modules/.env.
7. **eslint.config.js** : File konfigurasi ESLint untuk menjaga gaya penulisan kode konsisten serta mendeteksi error.
8. **index.html** : Template HTML utama aplikasi. React akan di-*inject* ke dalam <div id="root"> pada file ini.
9. **package.json** : File manifest proyek yang berisi metadata aplikasi, daftar dependency, dan script seperti dev, build, dan lint.
10. **package-lock.json** : File otomatis yang mengunci versi dependency agar hasil instalasi sama di semua komputer developer.
11. **README.md** : Dokumentasi proyek yang berisi penjelasan singkat, cara instalasi, dan cara menjalankan aplikasi.
12. **vercel.json** : Konfigurasi deployment jika aplikasi dihosting di Vercel, misalnya untuk pengaturan routing atau rewrites.
13. **vite.config.js** : File konfigurasi untuk Vite, build tool yang dipakai aplikasi. Biasanya berisi plugin, alias path, dan pengaturan build.

3. Konfigurasi

Konfigurasi pada *front-end* web pemesanan makanan online di kantin ini digunakan untuk menyimpan data environment serta mengatur pengaturan pada beberapa modul. Berikut adalah konfigurasi yang telah diterapkan.

Tabel 5. 3 Konfigurasi

Konfigurasi Environment (.env)	
	
Gambar 5. 2 Konfigurasi Penjual	
	
Gambar 5. 3 Konfigurasi Pembeli	
Deskripsi	
Pada aplikasi fronte-end PasMeal, terdapat beberapa pengaturan konfigurasi, antara lain: 1. VITE_API_URL: Variabel ini menyimpan alamat URL utama dari API <i>back-end</i>. <i>Front-end</i> menggunakan URL ini untuk semua permintaan data, seperti mengambil daftar menu, mengirim data pesanan, dan melakukan autentikasi pengguna. Dengan menyimpannya di sini, alamat <i>back-end</i> dapat dengan mudah diubah tanpa harus mengubah kode aplikasi secara langsung. 2. VITE_MIDTRANS_CLIENT_KEY: Ini adalah kunci klien untuk layanan payment gateway Midtrans. Kunci ini bersifat publik dan digunakan oleh <i>front-end</i> untuk menginisialisasi dan menampilkan antarmuka pembayaran Midtrans Snap, sehingga pengguna dapat melakukan transaksi secara aman.	

5.1.1 Product Backlog

Penyusunan product backlog pada aplikasi pemesanan makanan *online* di kantin dilakukan dengan menentukan prioritas setiap fitur agar pengembangan berjalan terarah. Rangkaian fitur yang diprioritaskan disajikan pada table berikut sebagai acuan tahapan pengembangan aplikasi.

Tabel 5. 4 Product Backlog

Kode	Use Case	Nama Backlog
PAS-FE.01	UC-01	Menampilkan daftar kios
PAS-FE.02	UC-02	Menampilkan daftar menu perkios
PAS-FE.03	UC-03	Menampilkan detail menu
PAS-FE.04	UC-04	Mencari menu
PAS-FE.05	UC-05	Mengelola keranjang
PAS-FE.06	UC-06	Membuat pesanan dari keranjang
PAS-FE.07	UC-07	Menyediakan Payment gateway
PAS-FE.08	UC-08	Menampilkan status dan estimasi waktu pesanan
PAS-FE.09	UC-09	Mengirim notifikasi pesanan selesai
PAS-FE.10	UC-10	Mengelola Menu
PAS-FE.11	UC-11	Mengirim notifikasi pesanan masuk

Kode	Use Case	Nama Backlog
PAS-FE.12	UC-12	Menampilkan daftar pesanan dan detail pesanan
PAS-FE.13	UC-13	Mengubah status pesanan
PAS-FE.14	UC-14	Menampilkan tipe pengantaran
PAS-FE.15		Autentikasi
PAS-FE.16		Beranda

5.1.2 Sprint Ke-1

Sprint pertama merupakan tahap awal dalam siklus pengembangan sistem, di mana aktivitas Scrum seperti perencanaan, implementasi, hingga evaluasi mulai dijalankan. Pada fase ini, tim berfokus pada pembangunan komponen dasar yang menjadi fondasi bagi pengembangan fitur berikutnya. Sprint ke-1 berlangsung selama 7 hari, yaitu pada periode 1 hingga 7 Agustus. Uraian berikut menjelaskan secara rinci tahapan yang dilakukan selama Sprint pertama.

5.1.2.1 Sprint Planning Ke-1

Sprint planning pertama berfokus pada pembangunan fondasi antarmuka pengguna (UI) beserta integrasinya. Tujuannya adalah menghadirkan halaman-halaman inti yang menjadi dasar interaksi pada aplikasi, meliputi autentikasi penjual untuk akses sistem, beranda sebagai tampilan awal, serta modul pengelolaan menu. Selain itu, pada tahap ini juga direncanakan implementasi tampilan daftar kios, daftar menu per kios, dan fitur pencarian menu.

Tabel 5. 5 Sprint Planning Ke-1

Kode	Nama Backlog	Estimasi Waktu	Prioritas
PAS-FE.15	Autentikasi (UI)	7 Hari	Tinggi
PAS-FE.10	Mengelola Menu (UI)		Tinggi
PAS-FE.16	Beranda (UI)		Tinggi
PAS-FE.01	Menampilkan daftar kios (UI)		Tinggi
PAS-FE.02	Menampilkan daftar menu perkios (UI)		Tinggi
PAS-FE.04	Mencari menu (UI)		Tinggi

5.1.2.2 Sprint Backlog ke-1

Sprint backlog Sprint 1 berisi tugas inti untuk membangun fondasi UI: autentikasi, beranda, daftar kios, daftar menu per kios dengan tombol tambah ke keranjang, pencarian menu, serta manajemen menu dasar. Pembagian backlog per komponen memudahkan penentuan prioritas dan pelacakan progres selama sprint.

Tabel 5. 6 Sprint Backlog Ke-1

Kode	Nama Backlog	Kebutuhan	Waktu Terpakai
PAS-FE.15	Autentikasi (UI)	Halaman registrasi penjual	7 Hari
		Halaman registrasi kios	
		Halaman verifikasi OTP	
		Halaman login	
		Halaman lupa kata sandi	
		Halaman reset kata sandi	
		Halaman profil	

Kode	Nama Backlog	Kebutuhan	Waktu Terpakai
PAS-FE.10	Mengelola Menu (UI)	Integrasi API untuk autentikasi	
		Halaman daftar menu	
		Halaman menambahkan menu	
		Halaman memperbarui menu	
		Membuat popup hapus menu	
		Membuat detail menu	
		Integrasi API untuk CRUD menu	
PAS-FE.16	Beranda (UI)	Membuat hero	
		Membuat kios card terbaru	
		Membuat card slider menu terbaru	
		Integrasi API untuk halaman beranda	
PAS-FE.01	Menampilkan daftar kios (UI)	Halaman kios	
		Membuat card kios	
		Mencari Kios	
		Integrasi kios	
PAS-FE.02	Menampilkan daftar menu perkios (UI)	Halaman menu	
		Membuat card menu	
		Integrasi menu	
PAS-FE.04	Mencari menu (UI)	Membuat search bar	
		Integrasi API untuk mencari menu	

5.1.2.3 Increment Ke-1

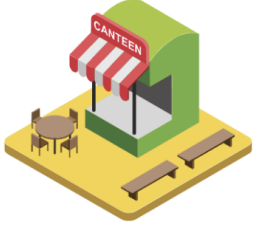
Pada akhir Sprint 1, sebuah increment fungsional yang menjadi fondasi utama *front-end* telah dihasilkan. Increment ini mencakup halaman-halaman dasar (autentikasi, daftar kios & menu, beranda, dan mengelola menu). Hasilnya terlihat pada struktur direktori proyek yang sudah terbentuk dan antarmuka awal tiap modul, yang siap dilanjutkan dengan logika bisnis dan integrasi API pada increment berikutnya.

1. Autentikasi (UI)

Antarmuka berikut merupakan hasil implementasi dari halaman autentikasi yang ditampilkan kepada penjual. Pada tampilan ini, komponen *Auth* memuat *Register Penjual*, *Register Toko*, Verifikasi OTP WhatsApp, *Login*, Lupa/Reset Sandi, dan Profil, yang membentuk alur dasar pembuatan akun hingga siap masuk *dashboard*.

Tabel 5. 7 Increment - Autentikasi (UI)

Tampilan Autentikasi



Jualan di kantin UNPAS?
Sekarang lebih praktis!

• • • •



Daftarkan akun Anda untuk melanjutkan

Nama

Nomor WhatsApp

Email

Kata Sandi

Konfirmasi Kata Sandi

Lanjut

Anda sudah memiliki akun? [Masuk](#)

Gambar 5. 4 Atarmuka Halaman Registrasi Penjual



Dukung pembayaran digital tanpa ribet

• • • •



Daftarkan Toko Anda untuk mulai penjualan

Nama Toko

Nama Bank

Nomor Rekening

Masuk

Gambar 5. 5 Atarmuka Halaman Registrasi Kios



Dukung pembayaran digital tanpa ribet

• • • •



Masukkan kode verifikasi yang dikirim ke nomor WhatsApp

Kode OTP

Tidak mendapatkan kode OTP? Kirim ulang 0257

Verifikasi

Gambar 5. 6 Atarmuka Halaman Verifikasi OTP



Jualan di kantin UNPAS?
Sekarang lebih praktis!



Silahkan masuk ke Akun Anda

Nama

Kata Sandi

☒ Ingatkan saya [Lupa Kata Sandi?](#)

Masuk

Tidak memiliki akun? [Daftar Disini](#)

Gambar 5. 7 Atarmuka Halaman Login



Jualan di kantin UNPAS?
Sekarang lebih praktis!



Masukkan nomor WhatsApp
Anda untuk atur ulang sandi

Nomor WhatsApp

Kirim

Gambar 5. 8 Atarmuka Halaman Lupa Kata Sandi



Perbarui menu kapan pun



Permintaan atur ulang kata sandi diterima.
Silakan masukkan kata sandi baru Anda

Kata Sandi Baru

Konfirmasi Kata Sandi Baru

Ubah Kata Sandi

Gambar 5. 9 Atarmuka Halaman Reset Kata Sandi

Gambar 5. 10 Atarmuka Edit Profil Penjual

Gambar 5. 11 Atarmuka Edit Profil Toko

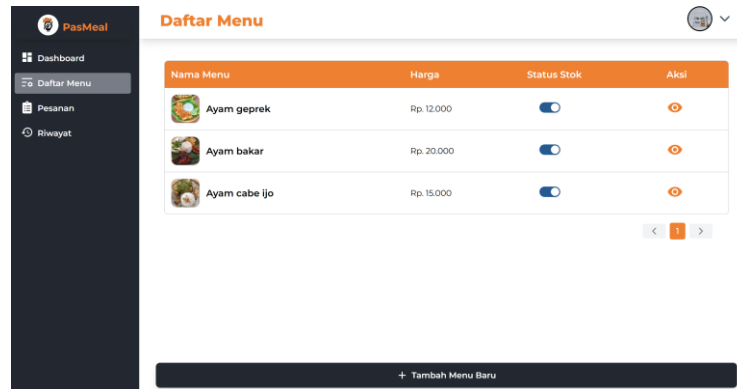
Deskripsi

Halaman ini merupakan tampilan autentikasi pengguna yang menyediakan formulir untuk masuk dan mendaftar. Pengguna baru membuat akun dengan mengisi nama, nomor WhatsApp, email, serta kata sandi (termasuk konfirmasi), lalu langsung mendaftarkan toko dengan melengkapi nama toko, bank, dan nomor rekening; setelah itu sistem mengirim kode OTP via WhatsApp untuk verifikasi (tersedia opsi kirim ulang dengan hitung mundur). Pengguna yang sudah terverifikasi dapat masuk menggunakan nama dan kata sandi, dengan opsi ingatlan saya serta tautan untuk berpindah antara masuk dan daftar. Tersedia juga fitur "lupa kata sandi" yang mengirim link reset kata sandi ke WhatsApp untuk mengatur ulang kata sandi, dan setelah masuk penjual dapat mengedit profil serta melengkapi profil toko di dashboard.

2. Mengelola Menu (UI)

Antarmuka berikut merupakan hasil implementasi dari mengelola menu (UI) di dashboard penjual. Pada tampilan ini, komponen daftar menu, tambah dan ubah menu, detail menu, dan popup konfirmasi hapus menu, unggah/ganti gambar, dan toggle ketersediaan disiapkan sebagai dasar CRUD sebelum integrasi API.

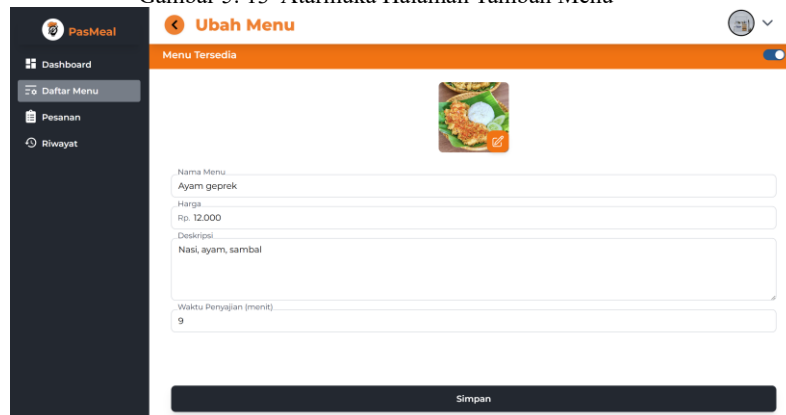
Tabel 5. 8 Increment - Mengelola Menu (UI)
Tampilan Mengelola Menu



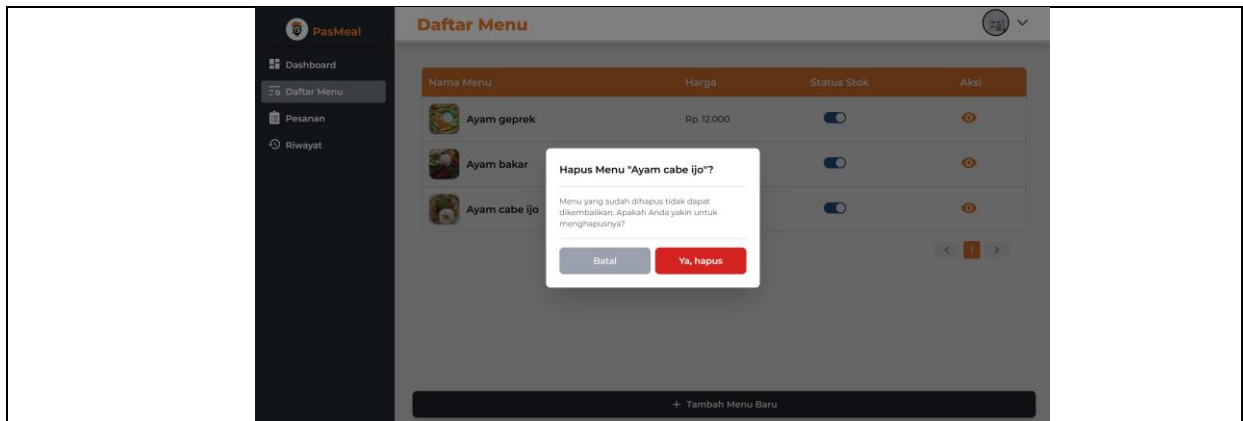
Gambar 5. 12 Atarmuka Halaman Daftar Menu



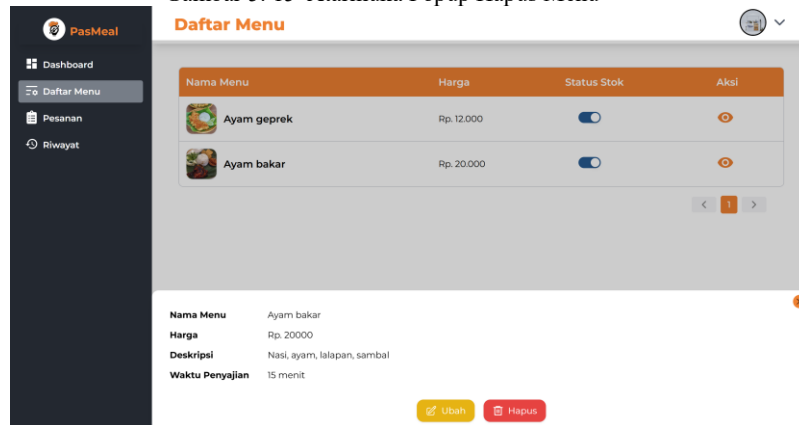
Gambar 5. 13 Atarmuka Halaman Tambah Menu



Gambar 5. 14 Atarmuka Halaman Edit Menu



Gambar 5. 15 Atarmuka Popup Hapus Menu



Gambar 5. 16 Atarmuka Detail Menu

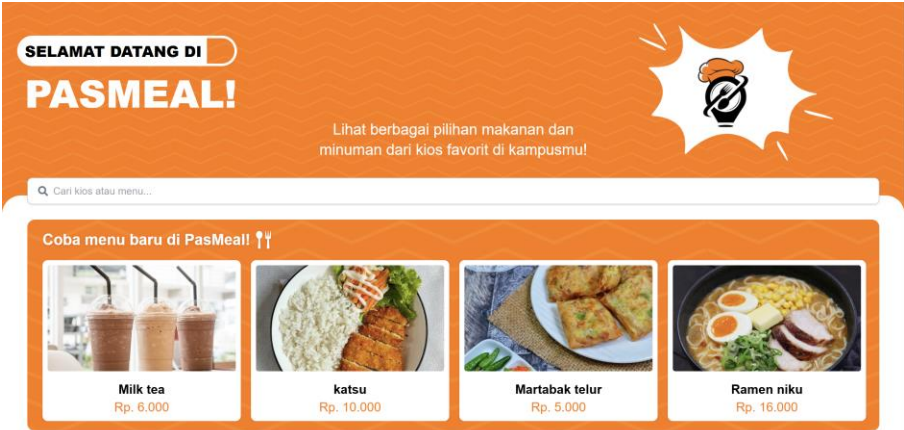
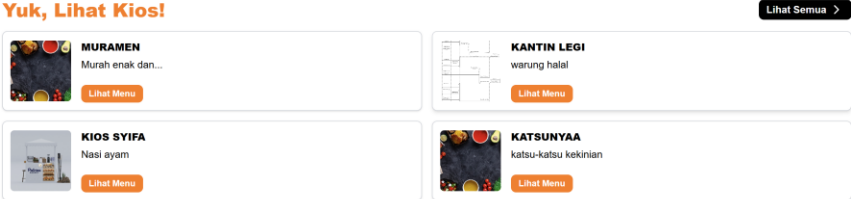
Deskripsi

Halaman Daftar Menu menampilkan produk beserta foto, nama, harga, dan status stok (toggle aktif/nonaktif) dengan tombol lihat detail, pagination, serta tombol Tambah Menu Baru; pada Buat Menu Baru penjual dapat unggah gambar lalu mengisi Nama Menu, Harga, Deskripsi, dan Waktu Penyajian (menit) serta menetapkan ketersediaan sebelum menekan Buat Menu; pada Ubah Menu tersedia form yang sama untuk edit data (termasuk ganti gambar) dan Simpan; penghapusan item memunculkan popup konfirmasi (Batal / Ya, hapus); dan panel Detail Menu menampilkan ringkasan (nama, harga, deskripsi, waktu penyajian) dengan aksi cepat Ubah dan Hapus.

3. Beranda (UI)

Antarmuka berikut merupakan hasil implementasi dari halaman beranda yang ditampilkan kepada pelanggan. Pada tampilan ini, komponen halaman beranda memuat bagian utama seperti Hero, *Slider* Menu Terbaru, dan Kios Terbaru yang membentuk struktur dasar halaman utama.

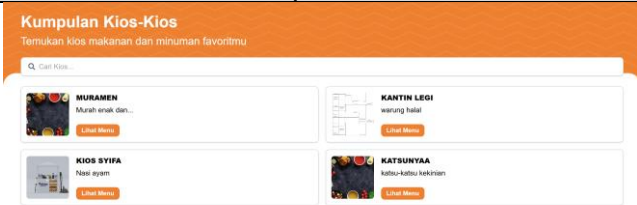
Tabel 5. 9 Increment - Beranda (UI)

Tampilan Beranda	
	
Gambar 5. 17 Atarmuka Beranda - Hero dan Card Slider	
	
Gambar 5. 18 Atarmuka Beranda - Kios Card	
Deskripsi	
Halaman ini adalah beranda utama dengan menampilkan menu dan kios terbaru.	

4. Menampilkan daftar kios (UI)

Antarmuka berikut merupakan hasil implementasi dari halaman daftar kios. Pada tampilan ini, komponen halaman kios memuat Search Bar dan KiosCard berisi foto, nama, deskripsi, serta tombol Lihat Menu untuk menuju halaman menu kios.

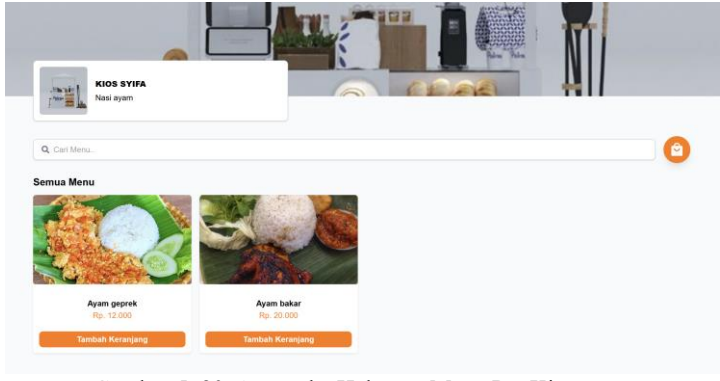
Tabel 5. 10 Increment - Menampilkan daftar kios (UI)

Tampilan Daftar Kios	
	
Gambar 5. 19 Atarmuka Halaman Kios	
Deskripsi	
Halaman Daftar Kios menampilkan kumpulan kios dengan kolom pencarian untuk memfilter cepat; setiap kartu berisi foto, nama kios, deskripsi singkat, dan tombol Lihat Menu yang mengarah ke halaman menu kios tersebut (detail kios) agar pengguna bisa melihat seluruh menu dan lanjut memesan.	

5. Menampilkan daftar menu perkios (UI)

Antarmuka berikut merupakan hasil implementasi dari halaman daftar menu per kios. Pada tampilan ini, komponen halaman menu menampilkan MenuCard (foto, nama, harga), tombol Tambah Keranjang untuk memasukkan item ke keranjang, serta ikon keranjang mengambang untuk meninjau pesanan.

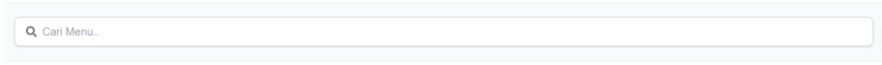
Tabel 5. 11 Increment - Menampilkan daftar menu perkios (UI)

Tampilan Daftar Menu	
	
Gambar 5. 20 Antarmuka Halaman Menu Per Kios	
Deskripsi	
Halaman Daftar Menu per Kios menampilkan detail kios (foto, nama, deskripsi), kolom pencarian menu untuk memfilter, serta setiap item disajikan sebagai kartu berisi foto, nama menu, harga, serta tombol Tambah Keranjang yang berfungsi untuk menambahkan menu tersebut ke keranjang, sementara ikon keranjang mengambang di kanan atas membuka ringkasan keranjang untuk meninjau dan melanjutkan proses pemesanan.	

6. Mencari menu (UI)

Antarmuka berikut merupakan hasil implementasi dari fitur mencari menu (UI). Pada tampilan ini, komponen SearchBar disediakan sebagai elemen antarmuka, saat ini berfungsi sebagai tampilan saja tanpa logika filter atau pemanggilan API.

Tabel 5. 12 Increment - Mencari menu (UI)

Tampilan Mencari Menu	
	
Gambar 5. 21 Antarmuka Search Bar	
Deskripsi	
Fitur Mencari Menu (UI saja) menampilkan kolom input "Cari Menu.." dengan ikon pencarian di bagian atas, saat ini hanya tampilan tanpa logika filter atau pemanggilan API, sehingga daftar menu tidak berubah saat pengguna mengetik.	

5.1.2.4 Sprint Review Ke-1

Sprint Review dilakukan pada akhir setiap iterasi sprint untuk meninjau serta memverifikasi Increment yang sudah selesai. Dalam sesi ini, setiap backlog item yang telah dikerjakan ditampilkan dan dievaluasi guna memperoleh masukan. *Feedback* yang dikumpulkan kemudian dimanfaatkan sebagai acuan dalam penyusunan rencana sprint berikutnya.

Tabel 5. 13 Sprint Review Ke-1

Kode	Nama Backlog	Status	Hasil dan Umpan Balik
PAS-FE.15	Autentikasi (UI)	Tidak Selesai	Hasil: Untuk tampilan halaman registrasi penjual, registrasi kios, verifikasi OTP, login, lupa kata sandi, reset kata sandi, dan profil telah berhasil dibuat Umpan Balik: Integrasi dengan API untuk semua alur autentikasi belum dikerjakan. Oleh karena itu, backlog in akan di jadwalkan ulang untuk diselesaikan pada sprint berikutnya.
PAS-FE.10	Mengelola Menu (UI)	Tidak Selesai	Hasil: Untuk tampilan halaman daftar menu, menambahkan menu, memperbarui menu, komponen popup hapus menu, dan detail menu telah berhasil dibuat Umpan Balik: Integrasi dengan API untuk semua alur mengelola menu belum dikerjakan. Oleh karena itu, backlog in akan di jadwalkan ulang untuk diselesaikan pada sprint berikutnya.
PAS-FE.16	Beranda (UI)	Tidak Selesai	Hasil: Untuk komponen antarmuka untuk halaman beranda (hero, kios card terbaru dan card slider menu baru) telah berhasil dibuat Umpan Balik: Integrasi dengan API untuk menampilkan menu dan kios baru belum dikerjakan. Oleh karena itu, backlog in akan di jadwalkan ulang untuk diselesaikan pada sprint berikutnya.
PAS-FE.01	Menampilkan daftar kios (UI)	Tidak Selesai	Hasil: Untuk tampilan halaman daftar kios dan komponen card kios telah berhasil dibuat Umpan Balik: Integrasi dengan API untuk menampilkan daftar kios dan mencari kios belum dikerjakan. Oleh karena itu, backlog in akan di jadwalkan ulang untuk diselesaikan pada sprint berikutnya.
PAS-FE.02	Menampilkan daftar menu perkios (UI)	Tidak Selesai	Hasil: Untuk tampilan halaman daftar menu dan komponen card menu telah berhasil dibuat Umpan Balik: Integrasi dengan API untuk menampilkan daftar menu perkios belum dikerjakan. Oleh karena itu, backlog in akan di jadwalkan ulang untuk diselesaikan pada sprint berikutnya.
PAS-FE.04	Mencari menu (UI)	Tidak Selesai	Hasil: Untuk komponen search bar telah berhasil dibuat Umpan Balik: Integrasi dengan API untuk mencari menu belum dikerjakan. Oleh karena itu, backlog in akan di jadwalkan ulang untuk diselesaikan pada sprint berikutnya.

5.1.3 Sprint Ke-2

Sprint kedua merupakan kelanjutan dari proses pengembangan sistem, yang dilaksanakan pada periode 8 hingga 14 Agustus dengan durasi 7 hari. Pada tahap ini, tim mulai mengerjakan backlog dengan prioritas tinggi yang berfokus pada integrasi serta penyempurnaan fungsi utama aplikasi. Aktivitas Scrum seperti perencanaan, implementasi, hingga evaluasi tetap dijalankan untuk memastikan setiap Increment sesuai tujuan.

5.1.3.1 Sprint Planning Ke-2

Sprint planning kedua berfokus pada integrasi dan penyempurnaan alur utama aplikasi di sisi klien. Tujuannya memastikan fitur inti mulai dari masuk, melihat kios dan menu, mencari, melihat detail, hingga menambahkan ke keranjang serta pengelolaan menu terhubung stabil ke API *back-end* dan siap digunakan pengguna secara mulus. Aktivitas Scrum (perencanaan, implementasi, evaluasi) tetap dijalankan untuk menjaga keselarasan tujuan dan kualitas hasil.

Tabel 5. 14 Sprint Planning Ke-2

Kode	Nama Backlog	Estimasi Waktu	Prioritas
PAS-FE.15	Autentikasi (Integrasi)	7 Hari	Tinggi
PAS-FE.10	Mengelola Menu (Integrasi)		Tinggi
PAS-FE.01	Menampilkan daftar kios		Tinggi
PAS-FE.02	Menampilkan daftar menu per kios		Tinggi
PAS-FE.04	Mencari Menu (Integrasi)		Tinggi
PAS-FE.03	Menampilkan detail menu (UI dan Integrasi)		Tinggi
PAS-FE.05	Mengelola keranjang (UI dan Integrasi)		Tinggi

5.1.3.2 Sprint backlog ke-2

Sprint backlog pada sprint kedua memuat tugas integrasi inti di sisi klien: menghubungkan autentikasi ke API, merampungkan CRUD kelola menu, memuat data daftar kios dan menu per kios, membangun halaman detail menu, mengaktifkan pencarian menu, serta pengelolaan keranjang. Pekerjaan diatur per halaman dan integrasi agar penyelesaiannya terstruktur dalam waktu sprint.

Tabel 5. 15 Sprint Backlog Ke-2

Kode	Nama Backlog	Kebutuhan	Waktu Terpakai
PAS-FE.15	Autentikasi (Integrasi)	Mengintegrasikan fungsi registrasi penjual dengan API <i>back-end</i> .	7 hari
		Mengintegrasikan fungsi registrasi kios dengan API <i>back-end</i> .	
		Mengimplementasikan fungsi verifikasi OTP dan kirim ulang OTP melalui API	
		Mengimplementasikan fungsi Lupa Kata Sandi dan Reset Kata Sandi dengan API.	
PAS-FE.10	Mengelola Menu (Integrasi)	Mengintegrasikan semua fungsi CRUD menu dengan API <i>back-end</i> .	
PAS-FE.01	Menampilkan daftar kios (Integrasi)	Mengintegrasikan fungsi menampilkan daftar kios dengan API <i>back-end</i> .	
PAS-FE.02	Menampilkan daftar menu per kios (Integrasi)	Mengintegrasikan fungsi menampilkan daftar menu perkios dengan API <i>back-end</i> .	
PAS-FE.04	Mencari menu (Integrasi)	Mengintegrasikan fungsi mencari menu dengan API <i>back-end</i> .	
PAS-FE.03	Menampilkan detail menu (UI dan Integrasi)	Halaman detail menu	
PAS-FE.05	Mengelola keranjang (UI dan Integrasi)	Integrasi API untuk menampilkan detail menu	
		Membuat komponen keranjang	
		Mengintegrasikan API untuk menambahkan, mengubah jumlah, dan menghapus menu pada keranjang.	

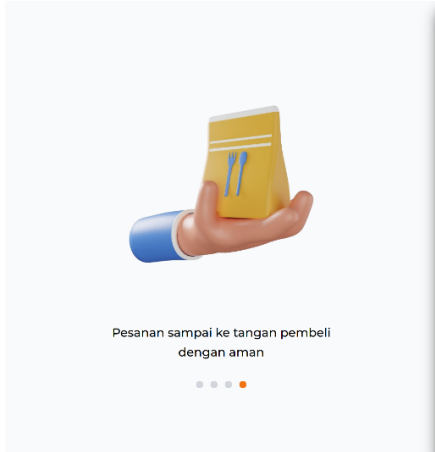
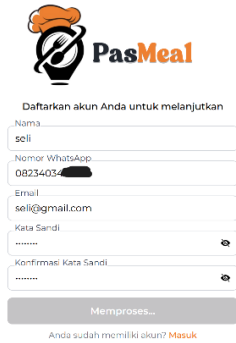
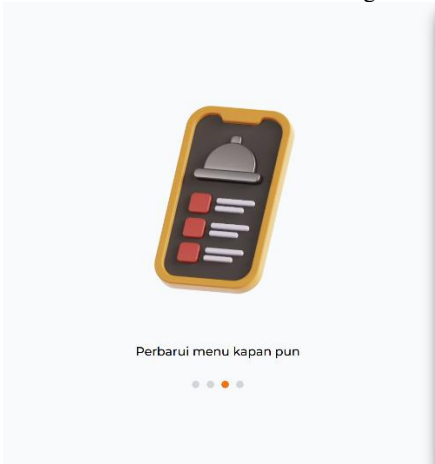
5.1.3.3 Increment Ke-2

Pada akhir Sprint 2, sebuah increment berhasil dihasilkan yang berfokus pada integrasi fungsi inti aplikasi. Increment ini mencakup autentikasi yang sudah terhubung dengan API *back-end*, pemuatan dinamis daftar kios beserta menu per kios, penambahan halaman detail menu, pengaktifan fitur pencarian menu, serta integrasi CRUD kelola menu. Selain itu, komponen UI keranjang telah selesai dibangun, meskipun integrasi penuh dengan API masih dalam proses penyempurnaan untuk sprint berikutnya.

1. Autentikasi (Integrasi)

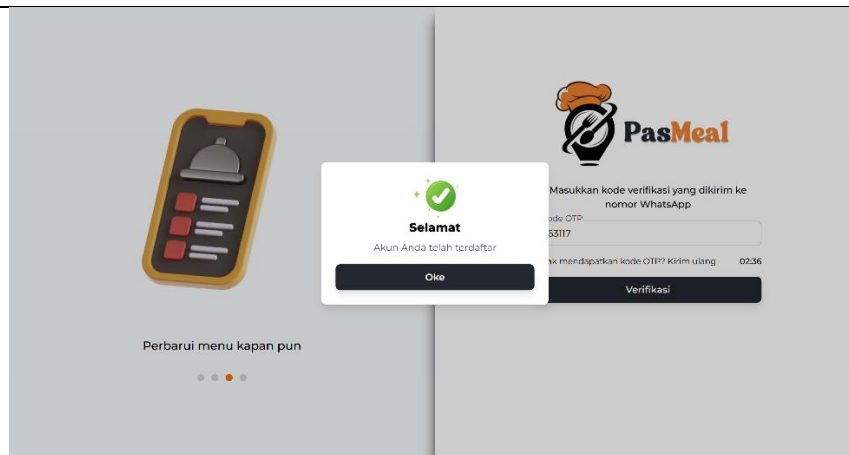
Antarmuka berikut merupakan hasil integrasi dari halaman autentikasi pengguna. Pada tampilan ini, pengguna dapat melakukan registrasi dengan melengkapi data diri dan toko, menerima OTP via WhatsApp untuk verifikasi, serta masuk menggunakan akun terverifikasi dengan dukungan fitur “ingat saya” dan “lupa kata sandi”.

Tabel 5. 16 Increment - Autentikasi (Integrasi)

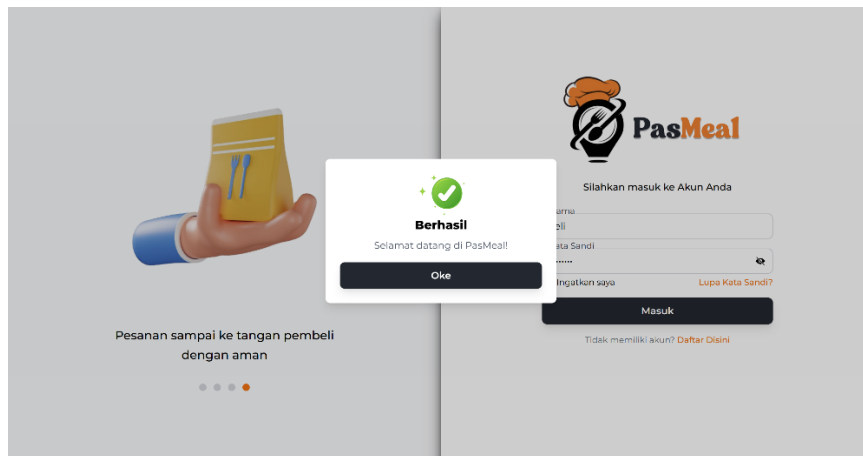
Tampilan Autentikasi (Integrasi)	
	
	

Gambar 5. 22 Integrasi Halaman Registrasi Penjual

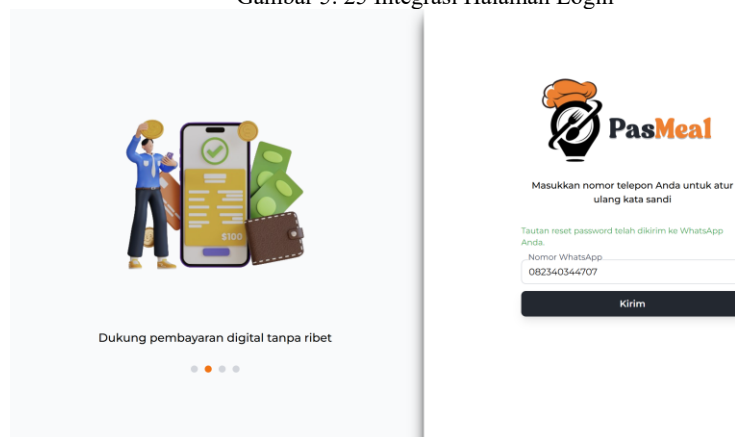
Gambar 5. 23 Integrasi Halaman Registrasi Kios



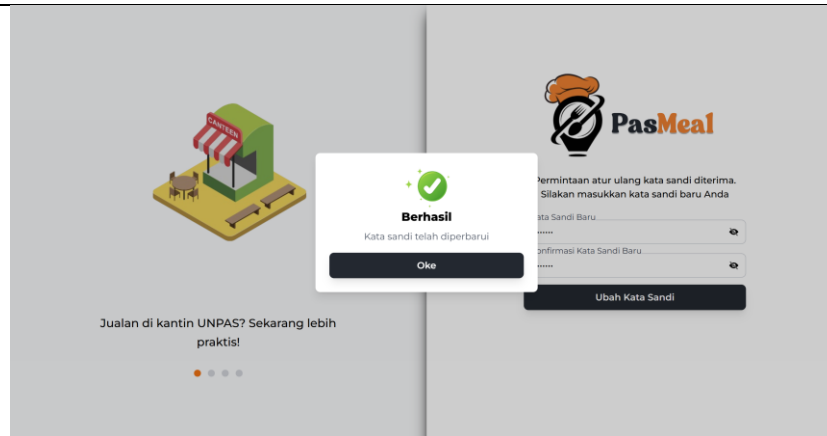
Gambar 5. 24 Integrasi Halaman Verifikasi OTP



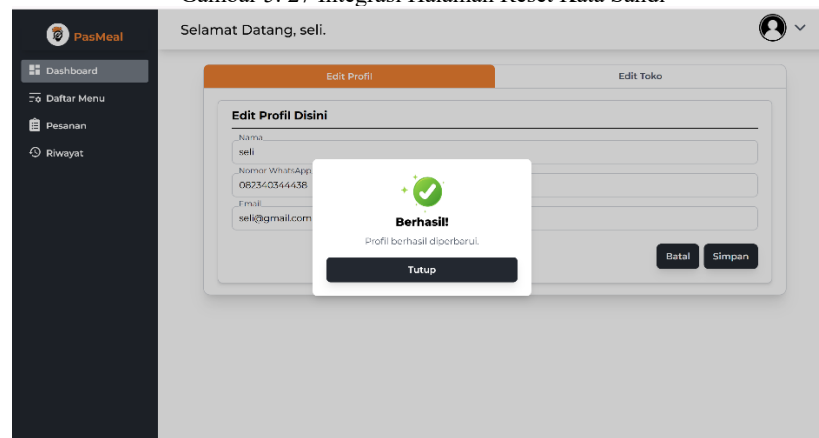
Gambar 5. 25 Integrasi Halaman Login



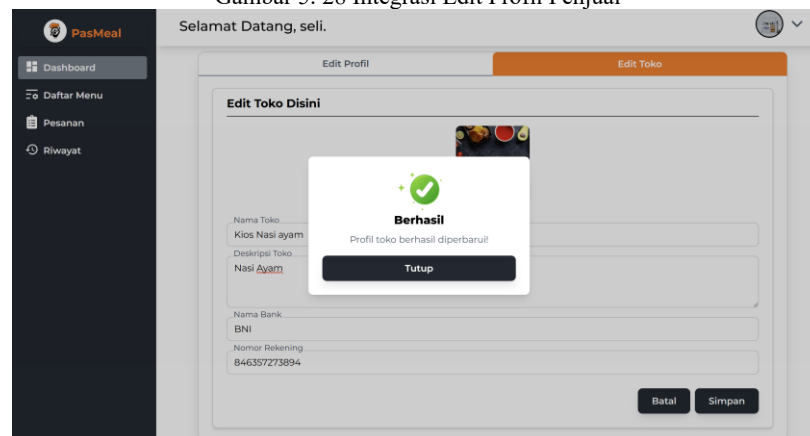
Gambar 5. 26 Integrasi Halaman Lupa Kata Sandi



Gambar 5. 27 Integrasi Halaman Reset Kata Sandi



Gambar 5. 28 Integrasi Edit Profil Penjual



Gambar 5. 29 Integrasi Halaman Edit Profil Toko

Deskripsi

Fitur autentikasi telah berhasil dihubungkan dengan server sehingga sudah dapat digunakan secara nyata. Proses registrasi tidak hanya menampilkan formulir, tetapi juga benar-benar menyimpan data pengguna dan toko ke dalam sistem. Setelah pendaftaran, kode OTP otomatis dikirim melalui WhatsApp untuk verifikasi akun dengan opsi kirim ulang. Login pun sudah terhubung dengan database, mendukung fitur “ingat saya” dan pemulihan akun lewat “lupa kata sandi” yang mengirimkan tautan reset ke WhatsApp. Setelah berhasil masuk, penjual langsung diarahkan ke dashboard dan dapat mengelola profil maupun toko dengan data yang tersimpan secara permanen di server.

2. Mengelola Menu (Integrasi)

Antarmuka berikut merupakan hasil integrasi dari halaman pengelolaan menu pada dashboard penjual. Penjual dapat menambah, mengubah, maupun menghapus menu, dan seluruh data tersimpan langsung ke server sehingga pembeli selalu melihat informasi menu terbaru.

Tabel 5. 17 Increment - Mengelola Menu (Integrasi)

Tampilan Mengelola Menu (Integrasi)

The screenshot shows the 'Daftar Menu' page in the PasMeal application. On the left is a dark sidebar with a menu containing 'Dashboard', 'Daftar Menu' (highlighted), 'Pesanan', and 'Riwayat'. The main content area has a title 'Daftar Menu' and a table with the following data:

Nama Menu	Harga	Status Stok	Aksi
Ayam cabe ijo	Rp. 16.000	☒	
Ayam Geprek	Rp. 10.000	☒	
Ayam Bakar Madu	Rp. 20.000	☒	

Below the table is a pagination bar showing '< 1 >'. At the bottom of the main area is a dark button labeled '+ Tambah Menu Baru'.

Gambar 5. 30 Integrasi Halaman Daftar Menu

This screenshot shows the 'Daftar Menu' page after a new menu item has been added. The table now contains 4 items:

Nama Menu	Harga	Status Stok	Aksi
Ayam cabe ijo	Rp. 16.000	☒	
Ayam Geprek	Rp. 10.000	☒	
Ayam Bakar Madu	Rp. 20.000	☒	

A green notification banner at the bottom of the table area reads 'Berhasil menambahkan menu'. The '+ Tambah Menu Baru' button remains at the bottom.

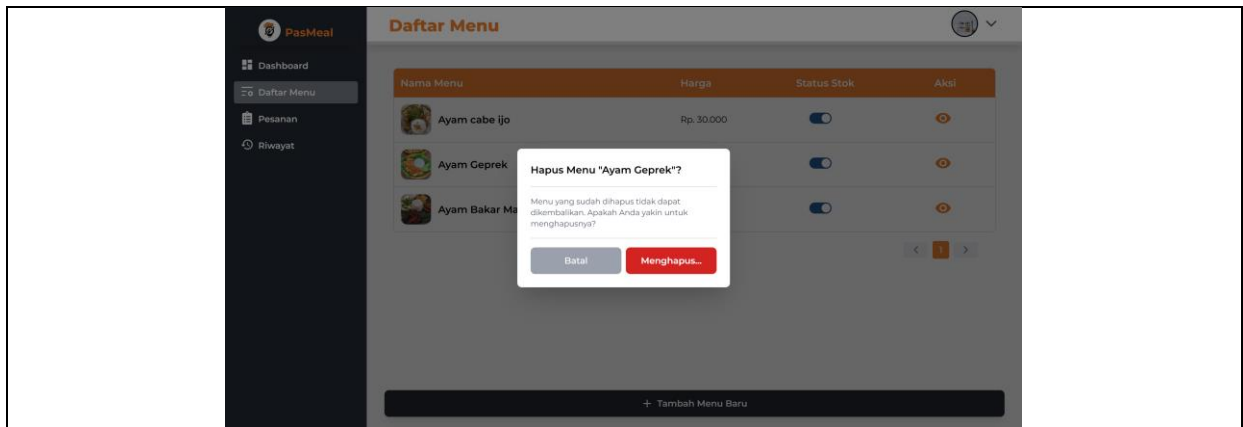
Gambar 5. 31 Integrasi Berhasil Menambahkan Menu Baru

This screenshot shows the 'Daftar Menu' page after an existing menu item has been updated. The table now contains 4 items, with the price of 'Ayam cabe ijo' changed to Rp. 30.000:

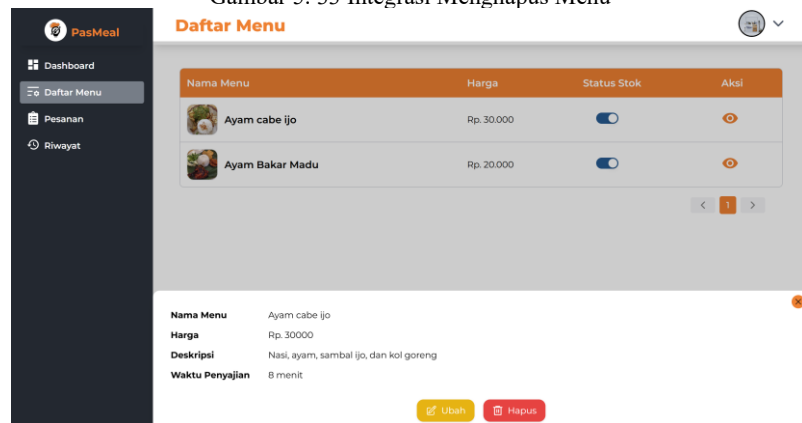
Nama Menu	Harga	Status Stok	Aksi
Ayam cabe ijo	Rp. 30.000	☒	
Ayam Geprek	Rp. 10.000	☒	
Ayam Bakar Madu	Rp. 20.000	☒	

A green notification banner at the bottom of the table area reads 'Berhasil mengubah menu'. The '+ Tambah Menu Baru' button remains at the bottom.

Gambar 5. 32 Integrasi Berhasil Mengubah Menu



Gambar 5. 33 Integrasi Menghapus Menu



Gambar 5. 34 Integrasi Detail Menu

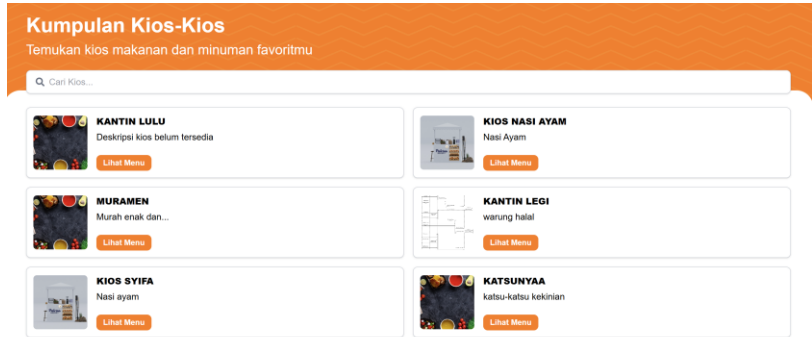
Deskripsi

laman ini merupakan hasil integrasi fitur pengelolaan menu, yang memungkinkan penjual menambah, mengubah, maupun menghapus menu secara langsung dari dashboard. Data menu yang dimasukkan penjual, seperti nama makanan, harga, deskripsi, dan foto, akan tersimpan ke server melalui API. Perubahan yang dilakukan juga otomatis diperbarui dan dapat langsung terlihat oleh pembeli pada aplikasi. Dengan demikian, penjual dapat mengatur ketersediaan menu secara real-time, sementara pembeli selalu mendapatkan informasi menu yang terbaru.

3. Menampilkan daftar kios (Integrasi)

Antarmuka berikut merupakan hasil integrasi dari halaman daftar kios. Data kios diambil dari server dan ditampilkan dalam bentuk kartu berisi nama, foto, dan status, sehingga pembeli dapat langsung memilih kios yang tersedia.

Tabel 5. 18 Increment - Menampilkan daftar kios (Integrasi)

Tampilan Menampilkan daftar kios (Integrasi)	
 Kumpulan Kios-Kios Temukan kios makanan dan minuman favoritmu Search: Cari Kios... KANTIN LULU Deskripsi kios belum tersedia Lihat Menu KIOS NASI AYAM Nasi Ayam Lihat Menu MURAMEN Murah enak dan... Lihat Menu KANTIN LEGI warung halat Lihat Menu KIOS SYIFA Nasi ayam Lihat Menu KATSUNYAA katsu-katsu kekinian Lihat Menu	
Gambar 5. 35 Integrasi Halaman Kios	

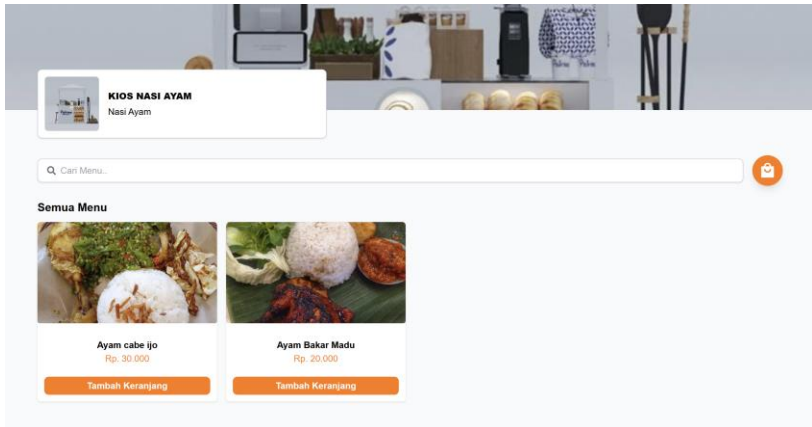
Deskripsi

Halaman ini menampilkan daftar kios yang telah terintegrasi dengan API, sehingga data yang muncul selalu sesuai dengan database terbaru. Setiap kios ditampilkan dalam bentuk kartu berisi informasi seperti nama kios, foto, dan status aktif. Data kios diambil secara langsung melalui API, sehingga apabila ada kios baru yang terdaftar atau perubahan pada profil kios, pembeli dapat segera melihat pembaruan tersebut di aplikasi tanpa perlu melakukan penyegaran manual.

4. Menampilkan daftar menu per kios (Integrasi)

Antarmuka berikut merupakan hasil integrasi dari halaman menu per kios. Informasi menu seperti nama, harga, dan foto diambil melalui API sehingga pembeli dapat melihat pilihan menu sesuai kios yang dipilih secara *real-time*.

Tabel 5. 19 Increment - Menampilkan daftar menu per kios (Integrasi)

Tampilan Menampilkan daftar menu per kios (Integrasi)	
 KIOS NASI AYAM Nasi Ayam Search: Cari Menu... Semua Menu Ayam cabe ijo Rp. 30.000 Tambah Keranjang Ayam Bakar Madu Rp. 20.000 Tambah Keranjang	
Gambar 5. 36 Integrasi Halaman Menu Per Kios	

Deskripsi

Halaman ini menampilkan daftar menu dari setiap kios yang sudah terhubung dengan API. Data menu seperti nama menu, harga, serta foto diambil melalui API sehingga informasi yang tampil selalu terbaru. Setiap kali penjual menambahkan, mengubah, atau menghapus menu, pembaruan tersebut langsung terlihat oleh pembeli pada halaman kios terkait. Dengan integrasi ini, pengguna dapat melihat pilihan menu secara lengkap sesuai kios yang dipilih tanpa perlu keluar dari aplikasi.

5. Menampilkan detail menu (UI dan Integrasi)

Antarmuka berikut merupakan hasil integrasi dari halaman detail menu. Informasi menu yang ditampilkan meliputi nama, harga, deskripsi, dan foto, dengan data yang selalu diperbarui dari server.

Tabel 5. 20 Increment - Menampilkan detail menu (UI dan Integrasi)

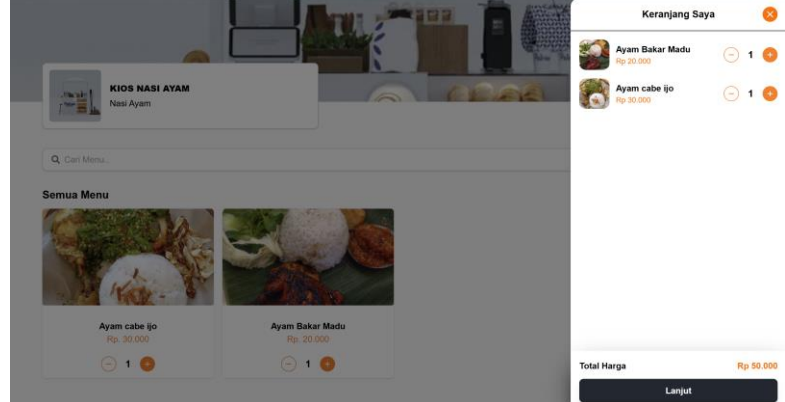
Tampilan Detail Menu

Gambar 5. 37 Atarmuka dan Integrasi Halaman Detail Menu Deskripsi Halaman ini menampilkan detail lengkap dari setiap menu yang dipilih pengguna, mulai dari nama menu, harga, deskripsi, dan foto. Data yang ditampilkan diambil langsung dari server melalui API, sehingga selalu sesuai dengan kondisi terbaru di database. Dengan integrasi ini, pengguna tidak hanya melihat tampilan UI, tetapi juga benar-benar mendapatkan informasi yang akurat mengenai menu, termasuk jika ada perubahan harga atau ketersediaan yang dilakukan oleh penjual

6. Mengelola keranjang (UI dan Integrasi)

Antarmuka berikut merupakan hasil integrasi dari halaman keranjang belanja. Pengguna dapat menambahkan item, mengubah jumlah, maupun menghapus menu dari keranjang, dengan perhitungan total harga yang diperbarui otomatis dan data tersimpan di server.

Tabel 5. 21 Increment - Mengelola keranjang (UI dan Integrasi)

Tampilan Keranjang

Gambar 5. 38 Atarmuka dan Integrasi Keranjang Deskripsi Halaman ini mendukung fitur pengelolaan keranjang belanja yang telah terintegrasi dengan server. Pengguna dapat menambahkan menu ke dalam keranjang, mengubah jumlah pesanan, maupun menghapus item, dan setiap perubahan langsung tersimpan di database sehingga data keranjang tetap konsisten meskipun berpindah halaman. Informasi total harga juga diperbarui secara otomatis sesuai isi keranjang.

5.1.3.4 Sprint Review Ke-2

Sprint Review ke-2 dilakukan pada akhir iterasi sprint kedua untuk meninjau serta memverifikasi increment yang telah diselesaikan. Dalam sesi ini, backlog item yang sudah berhasil diintegrasikan ditampilkan dan dievaluasi guna memperoleh masukan. Feedback yang diperoleh kemudian dimanfaatkan sebagai acuan dalam penyusunan rencana pada sprint selanjutnya.

Tabel 5. 22 Sprint Review Ke-2

Kode	Nama Backlog	Status	Hasil dan Umpan Balik
PAS-FE.15	Autentikasi (Integrasi)	Selesai	Hasil: Fitur ini dinyatakan Selesai karena seluruh alur autentikasi telah berhasil diimplementasikan dan diintegrasikan dengan API <i>back-end</i> . Umpan Balik: Fitur berjalan sesuai ekspektasi. Tidak ada perubahan yang diperlukan.
PAS-FE.10	Mengelola Menu (Integrasi)	Selesai	Hasil: Fitur ini dinyatakan Selesai karena seluruh alur CRUD menu telah berhasil diimplementasikan dan diintegrasikan dengan API <i>back-end</i> . Umpan Balik: Fitur berjalan sesuai ekspektasi. Tidak ada perubahan yang diperlukan.
PAS-FE.01	Menampilkan daftar kios (Integrasi)	Selesai	Hasil: Fitur ini dinyatakan selesai karena daftar kios berhasil ditampilkan secara dinamis dengan data dari API <i>back-end</i> . Umpan Balik: Fitur berjalan sesuai ekspektasi. Tidak ada perubahan yang diperlukan.
PAS-FE.02	Menampilkan daftar menu per kios (Integrasi)	Selesai	Hasil: Fitur ini dinyatakan selesai karena daftar menu per kios berhasil ditampilkan secara dinamis dengan data dari API <i>back-end</i> . Umpan Balik: Fitur berjalan sesuai ekspektasi. Tidak ada perubahan yang diperlukan.
PAS-FE.04	Mencari menu (Integrasi)	Selesai	Hasil: Fitur ini dinyatakan selesai karena fungsi pencarian menu berhasil menampilkan hasil sesuai kata kunci secara dinamis dengan data dari API <i>back-end</i> . Umpan Balik: Fitur berjalan sesuai ekspektasi. Tidak ada perubahan yang diperlukan.
PAS-FE.03	Menampilkan detail menu (UI dan Integrasi)	Selesai	Hasil: Fitur ini dinyatakan selesai karena halaman detail menu berhasil dibuat dan telah terintegrasi dengan API <i>back-end</i> untuk menampilkan informasi detail menu secara dinamis. Umpan Balik: Fitur berjalan sesuai ekspektasi. Tidak ada perubahan yang diperlukan.
PAS-FE.05	Mengelola keranjang (UI dan Integrasi)	Tidak Selesai	Hasil: Komponen keranjang sudah berhasil dibuat, namun integrasi dengan API belum berfungsi sepenuhnya. Keranjang pada halaman detail menu belum dapat menambahkan produk dengan benar. Umpan Balik: Integrasi API untuk fitur keranjang perlu diperbaiki agar fungsi penambahan produk dari halaman detail menu dapat berjalan. Backlog ini akan dijadwalkan ulang untuk sprint berikutnya.

5.1.4 Sprint Ke-3

Sprint ketiga merupakan lanjutan dari proses pengembangan sistem yang dilaksanakan pada periode 15 hingga 21 Agustus dengan durasi 7 hari. Pada tahap ini, tim berfokus pada pengembangan alur pemesanan dan transaksi, mulai dari pengelolaan keranjang, pembuatan pesanan, hingga pembayaran serta status pesanan. Aktivitas Scrum seperti perencanaan, implementasi, hingga evaluasi tetap dijalankan untuk memastikan setiap increment yang dihasilkan sesuai dengan kebutuhan dan tujuan sprint.

5.1.4.1 Sprint Planning Ke-3

Sprint planning ketiga berfokus pada pengembangan alur pemesanan secara menyeluruh, mulai dari pengelolaan keranjang hingga proses transaksi dan pelacakan pesanan. Tujuannya adalah memastikan pengguna dapat melakukan pembelian dengan lancar, mulai dari memilih menu, menambahkan ke keranjang, memilih tipe pengantaran, melakukan pembayaran melalui *payment gateway*, hingga memantau status dan estimasi waktu pesanan. Selain itu, pada sisi penjual juga dikembangkan fitur untuk melihat daftar pesanan, mengubah status pesanan, serta menerima notifikasi pesanan masuk.

Tabel 5. 23 Sprint Planning Ke-3

Kode	Nama Backlog	Estimasi Waktu	Prioritas
PAS-FE.16	Beranda (Integrasi)	7 Hari	Tinggi
PAS-FE.05	Mengelola keranjang (Integrasi)		Tinggi
PAS-FE.06	Membuat pesanan dari keranjang (UI dan Integrasi)		Tinggi
PAS-FE.14	Menampilkan tipe pengantaran (Integrasi)		Tinggi
PAS-FE.07	Menyediakan <i>Payment gateway</i> (UI dan Integrasi)		Tinggi
PAS-FE.12	Menampilkan daftar pesanan dan detail pesanan (UI dan Integrasi)		Tinggi
PAS-FE.08	Menampilkan status dan estimasi waktu pesanan (UI dan Integrasi)		Tinggi
PAS-FE.13	Mengubah status pesanan (UI dan Integrasi)		Tinggi
PAS-FE.11	Mengirim notifikasi pesanan masuk (Integrasi)		Tinggi

5.1.4.2 Sprint Backlog Ke-3

Sprint backlog pada sprint ketiga memuat tugas yang berfokus pada alur pemesanan dan transaksi pengguna. Pekerjaan mencakup integrasi halaman beranda, penyempurnaan fitur keranjang, pembuatan pesanan dari keranjang, serta penambahan opsi tipe pengantaran. Selain itu, tim juga mengerjakan integrasi *payment gateway* untuk mendukung proses pembayaran. Pada sisi pesanan, backlog terdiri dari penampilan daftar dan detail pesanan, pelacakan status serta estimasi waktu pesanan, pengubahan status pesanan oleh penjual, hingga pengiriman notifikasi pesanan masuk. Seluruh pekerjaan diatur agar penyelesaiannya terstruktur dalam lingkup sprint ketiga.

Tabel 5. 24 Sprint Backlog Ke-3

Kode	Nama Backlog	Kebutuhan	Waktu Terpakai
PAS-FE.16	Beranda (Integrasi)	Mengintegrasikan fungsi untuk menampilkan menu terbaru dengan API <i>back-end</i> .	7 hari
		Mengintegrasikan untuk menampilkan kios terbaru dengan API <i>back-end</i> .	
		Mengintegrasikan fungsi mencari menu dan kios dengan API <i>back-end</i>	
PAS-FE.05	Mengelola keranjang (Integrasi)	Mengintegrasikan fungsi keranjang dengan API <i>back-end</i> untuk menambahkan, mengubah jumlah, dan menghapus menu.	
PAS-FE.06	Membuat pesanan dari keranjang (UI dan Integrasi)	Halaman konfirmasi pesanan.	
		Integrasi API untuk membuat pesanan.	
PAS-FE.14	Menampilkan tipe pengantaran (Integrasi)	Mengintegrasikan tampilan tipe pengantaran dengan API <i>back-end</i> untuk menampilkan informasi metode pengantaran secara dinamis.	
PAS-FE.07	Menyediakan <i>Payment gateway</i>	Mengintegrasikan halaman konfirmasi pesanan dengan Midtrans Snap untuk memproses pembayaran.	
PAS-FE.12		Halaman daftar pesanan.	

Kode	Nama Backlog	Kebutuhan	Waktu Terpakai
	Menampilkan daftar pesanan dan detail pesanan (UI dan Integrasi)	Halaman detail pesanan. Integrasi API untuk menampilkan daftar pesanan dan detail pesanan.	
PAS-FE.08	Menampilkan status dan estimasi waktu pesanan (UI dan Integrasi)	Halaman status pesanan. Integrasi API status pesanan dan estimasi waktu pesanan.	
PAS-FE.13	Mengubah status pesanan (UI dan Integrasi)	Membuat komponen status pesanan. Integrasi API untuk mengubah status pesanan.	
PAS-FE.11	Mengirim notifikasi pesanan masuk (UI dan Integrasi)	Membuat tampilan badge notifikasi pada sidebar pesanan untuk menandai adanya pesanan baru. Menghubungkan badge dengan API pesanan masuk agar jumlah pesanan baru ditampilkan dan diperbarui secara dinamis.	

5.1.4.3 Increment Ke-3

Pada akhir Sprint 3, sebuah increment berhasil dihasilkan yang memperkuat integrasi fitur utama aplikasi. Increment ini mencakup halaman beranda yang sudah terhubung dengan API, pengelolaan keranjang yang terintegrasi, penentuan tipe pengantaran, serta pengiriman notifikasi pesanan masuk. Hasilnya menunjukkan alur pemesanan dasar sudah berfungsi, dan sistem siap dilanjutkan dengan penyempurnaan fitur transaksi seperti pembuatan pesanan, payment gateway, serta detail dan status pesanan pada sprint berikutnya.

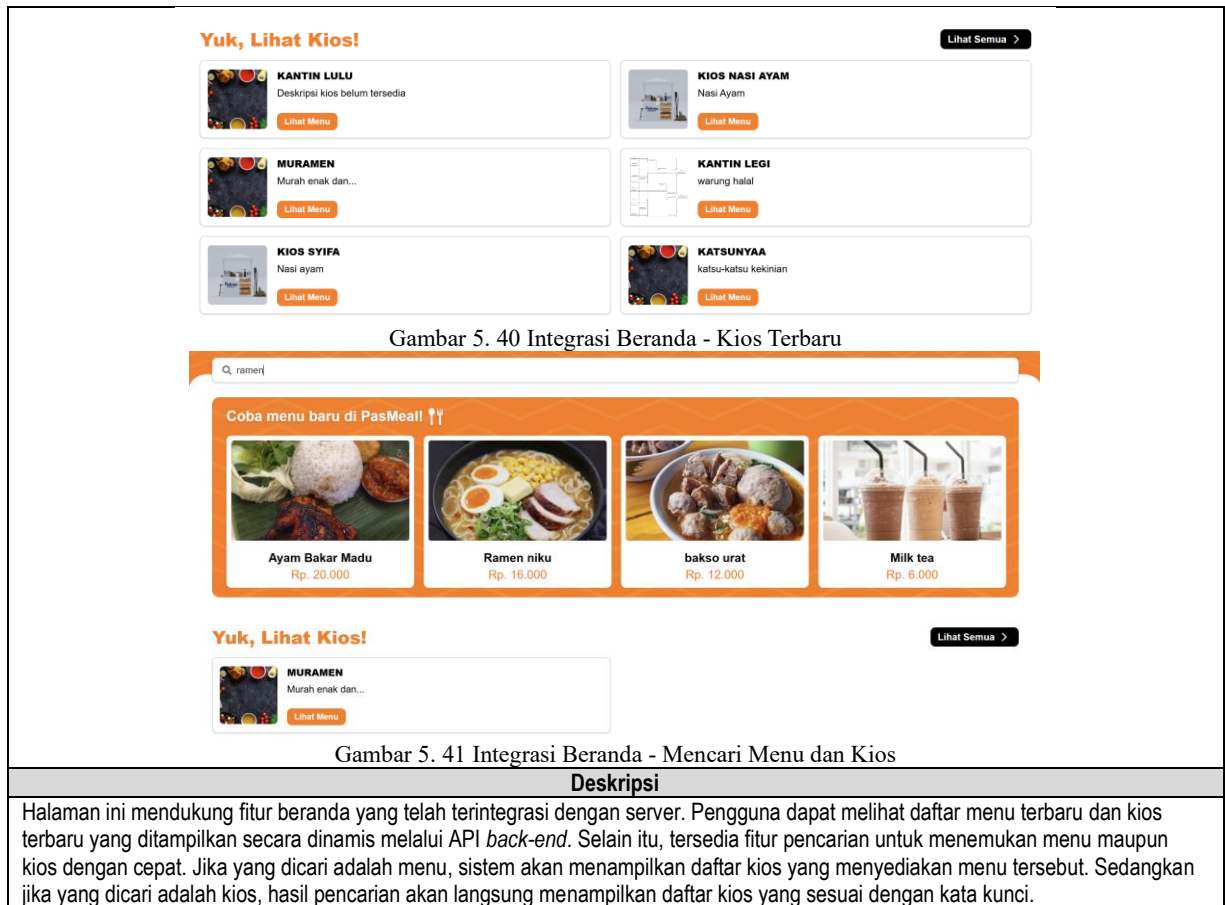
1. Beranda (Integrasi)

Antarmuka berikut merupakan hasil implementasi dari halaman beranda pada sisi pengguna. Pada sprint ini, pengembangan difokuskan untuk menampilkan daftar menu terbaru dan kios terbaru secara dinamis melalui API *back-end*. Selain itu, tersedia fitur pencarian yang memungkinkan pengguna menemukan kios berdasarkan menu yang dicari atau langsung mencari kios sesuai kata kunci.

Tabel 5. 25 Increment - Beranda (Integrasi)

Tampilan Beranda (Integrasi)

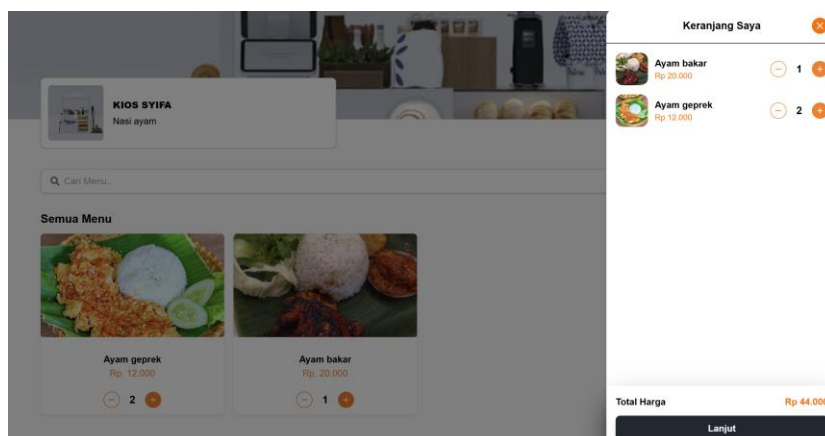
Gambar 5. 39 Integrasi Beranda - Card Slider Menu Terbaru



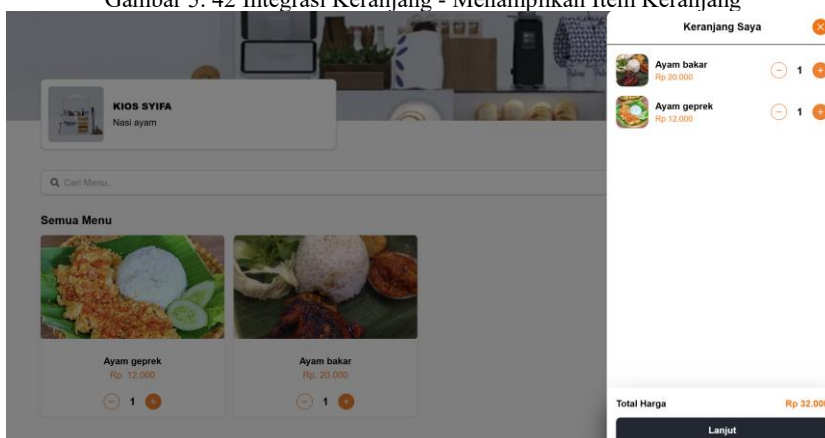
2. Mengelola keranjang (Integrasi)

Antarmuka berikut merupakan hasil implementasi dari fitur pengelolaan keranjang pada sisi pengguna. Pada sprint ini, pengembangan difokuskan untuk memungkinkan pengguna menambahkan menu ke dalam keranjang, mengubah jumlah pesanan, maupun menghapus item. Setiap perubahan langsung tersimpan di server sehingga data keranjang tetap konsisten, dan informasi total harga diperbarui secara otomatis sesuai isi keranjang.

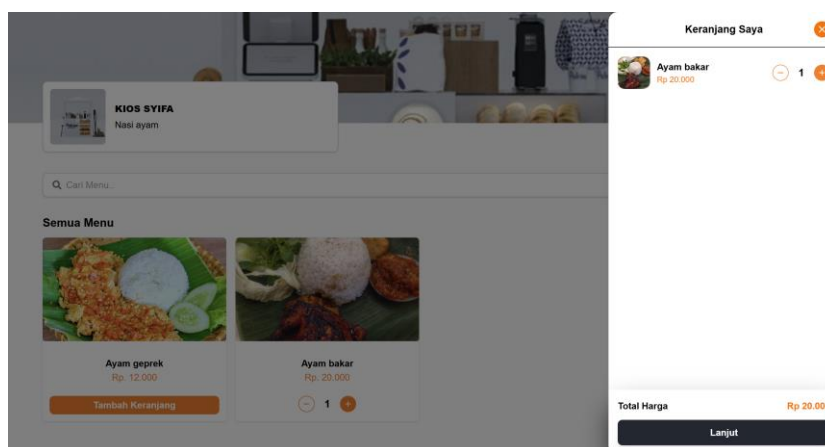
Tabel 5. 26 Increment - Mengelola keranjang (Integrasi)
Tampilan Keranjang (Integrasi)



Gambar 5. 42 Integrasi Keranjang - Menampilkan Item Keranjang



Gambar 5. 43 Integrasi Keranjang - Mengubah Jumlah Item Keranjang



Gambar 5. 44 Integrasi Keranjang - Menghapus Item Keranjang

Deskripsi

Halaman ini mendukung fitur pengelolaan keranjang belanja yang telah terintegrasi dengan server. Pengguna dapat menambahkan menu ke dalam keranjang, mengubah jumlah pesanan, maupun menghapus item, dan setiap perubahan langsung tersimpan di database. Informasi total harga juga diperbarui secara otomatis sesuai isi keranjang.

3. Membuat pesanan dari keranjang (UI dan Integrasi)

Antarmuka berikut merupakan hasil implementasi alur “Membuat Pesanan dari Keranjang”. Pengguna dapat menambahkan menu ke keranjang, meninjau ringkasan melalui ikon keranjang, mengubah jumlah atau menghapus item, lalu masuk ke Konfirmasi Pesanan untuk melengkapi data pemesan (nama, nomor HP, tipe pengantaran, catatan, dantar ke) dan melihat perhitungan subtotal serta total sebelum menekan Bayar.

Tabel 5. 27 Increment - Membuat pesanan dari keranjang (UI dan Integrasi)

Tampilan Konfirmasi Detail Pesanan	

Gambar 5. 45 Antarmuka dan Integrasi Halaman Konfirmasi Detail Pesanan

Deskripsi

Halaman Konfirmasi Pesanan menampilkan ringkasan keranjang berisi foto, nama menu, jumlah, harga, dan subtotal per item dengan kontrol tambah/kurang/hapus yang memperbarui total otomatis, dilengkapi formulir data pemesan (nama, nomor HP), pilihan tipe pengantaran (ambil sendiri/pesanan antar) beserta kolom catatan dan kolom “Diantar ke” yang muncul hanya jika memilih Pesanan antar, serta tombol Bayar untuk mengirim pesanan setelah pengguna meninjau detail dan total pembayaran.

4. Menampilkan tipe pengantaran (Integrasi)

Antarmuka sisi penjual menampilkan Tipe Pengantaran setiap pesanan secara dinamis dari API. Jika ambil sendiri, label yang ditampilkan adalah “Ambil sendiri”; jika diantar, label menampilkan lokasi tujuan sesuai alamat yang diinput pembeli pada halaman konfirmasi pesanan. Informasi ini ditampilkan pada Detail Pesanan agar penjual langsung mengetahui cara pemenuhan pesanan.

Tabel 5. 28 Increment - Menampilkan tipe pengantaran (Integrasi)

Tampilan Menampilkan tipe pengantaran (Integrasi)	

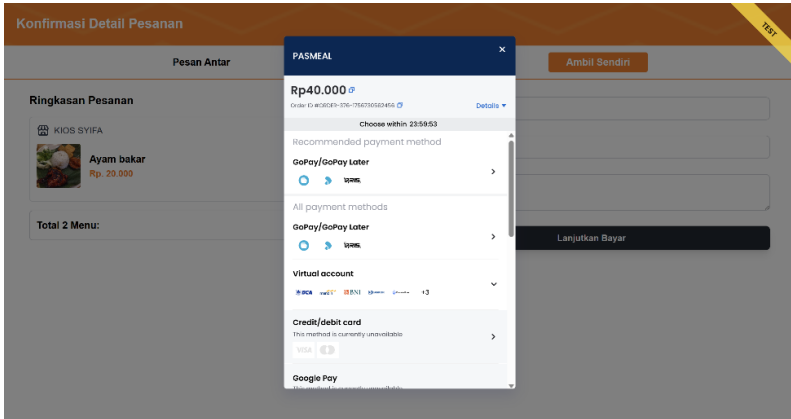
Gambar 5. 46 Integrasi Menampilkan Tipe Pengantaran

Deskripsi
Komponen Tipe Pengantaran (Integrasi) mengambil tipe_pengantaran dan diantar_ke dari <i>back-end</i> lalu merendernya pada halaman Detail Pesanan. Jika Ambil sendiri, yang ditampilkan adalah label “Ambil sendiri”; jika Diantar, yang ditampilkan adalah lokasi tujuan sesuai alamat yang diinput pembeli pada halaman konfirmasi.

5. Menyediakan Payment gateway

Antarmuka berikut menampilkan hasil implementasi dan integrasi payment gateway pihak ketiga dalam alur checkout. Saat pengguna menekan “Bayar”, aplikasi memanggil API payment gateway untuk memperoleh token transaksi lalu menampilkan modal pembayaran dengan metode yang dimuat secara dinamis dari layanan tersebut.

Tabel 5. 29 Increment - Menyediakan Payment gateway

Tampilan Payment gateway

Gambar 5. 47 Antarmuka <i>Payment Gateway</i>
Deskripsi
Antarmuka ini adalah pop-up (modal) dari payment gateway yang muncul setelah pengguna siap membayar. Di sini, pengguna melihat konfirmasi total akhir yang harus dibayarkan beserta batas waktu penyelesaian transaksi. Pengguna dapat memilih berbagai metode pembayaran yang disediakan, mulai dari virtual account berbagai bank hingga e-wallet.

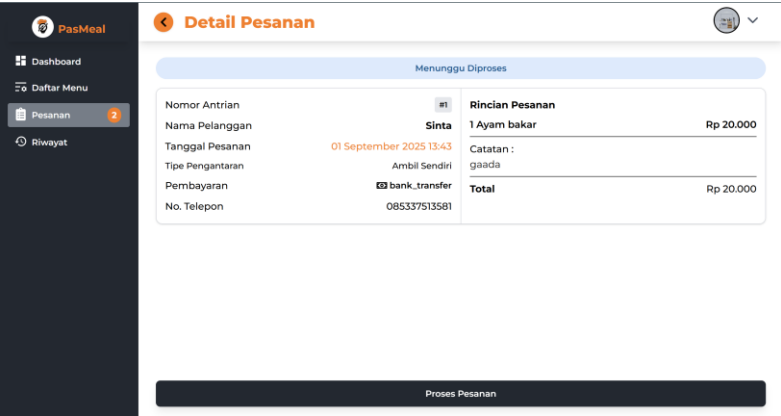
6. Menampilkan daftar pesanan dan detail pesanan (UI dan Integrasi)

Antarmuka Pesanan di sisi penjual menampilkan daftar pesanan yang sudah terintegrasi dengan *back-end* dan tautan ke Detail Pesanan. Dari daftar, penjual dapat membuka detail untuk meninjau informasi lengkap sebelum menekan Proses Pesanan.

Tabel 5. 30 Increment - Menampilkan daftar pesanan dan detail pesanan (UI dan Integrasi)
Tampilan Daftar Pesanan dan Detail Pesanan



Gambar 5. 48 Antarmuka dan Integrasi Halaman Daftar Pesanan



Gambar 5. 49 Antarmuka dan Integrasi Halaman Detail Pesanan

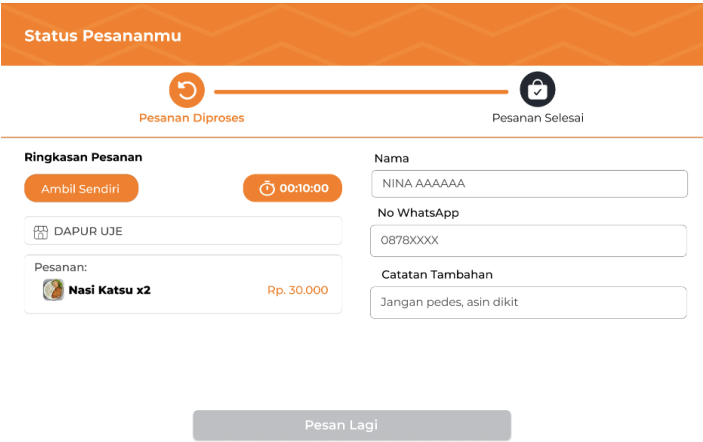
Deskripsi

Halaman Daftar Pesanan merender tabel berisi No, Nama Pelanggan, Tanggal, Harga, Status Pesanan, dan tombol Detail (ikon mata). Pada Detail Pesanan, tampil status pesanan, Nomor Antrian, Nama Pelanggan, Tanggal Pesanan, Tipe Pengantaran (Ambil Sendiri/Diantar ke), Pembayaran, No. Telepon, serta panel Rincian Pesanan yang memuat daftar menu, catatan, dan Total. Tombol Proses Pesanan disediakan untuk mengubah status pesanan.

7. Menampilkan status dan estimasi waktu pesanan (UI dan Integrasi)

Antarmuka Status Pesananmu menampilkan progres pesanan yang sedang berjalan. Pengguna dapat melihat tahap Pesanan Diproses, Pesanan Selesai, estimasi waktu melalui penghitung mundur, jenis pengantaran (Ambil Sendiri/Pesan Antar), serta ringkasan pesanan berisi kios, item, dan harga.

Tabel 5. 31 Increment - Menampilkan status dan estimasi waktu pesanan (UI dan Integrasi)
Tampilan Status Pesanan



Gambar 5. 50 Antarmuka Halaman Status Pesanan

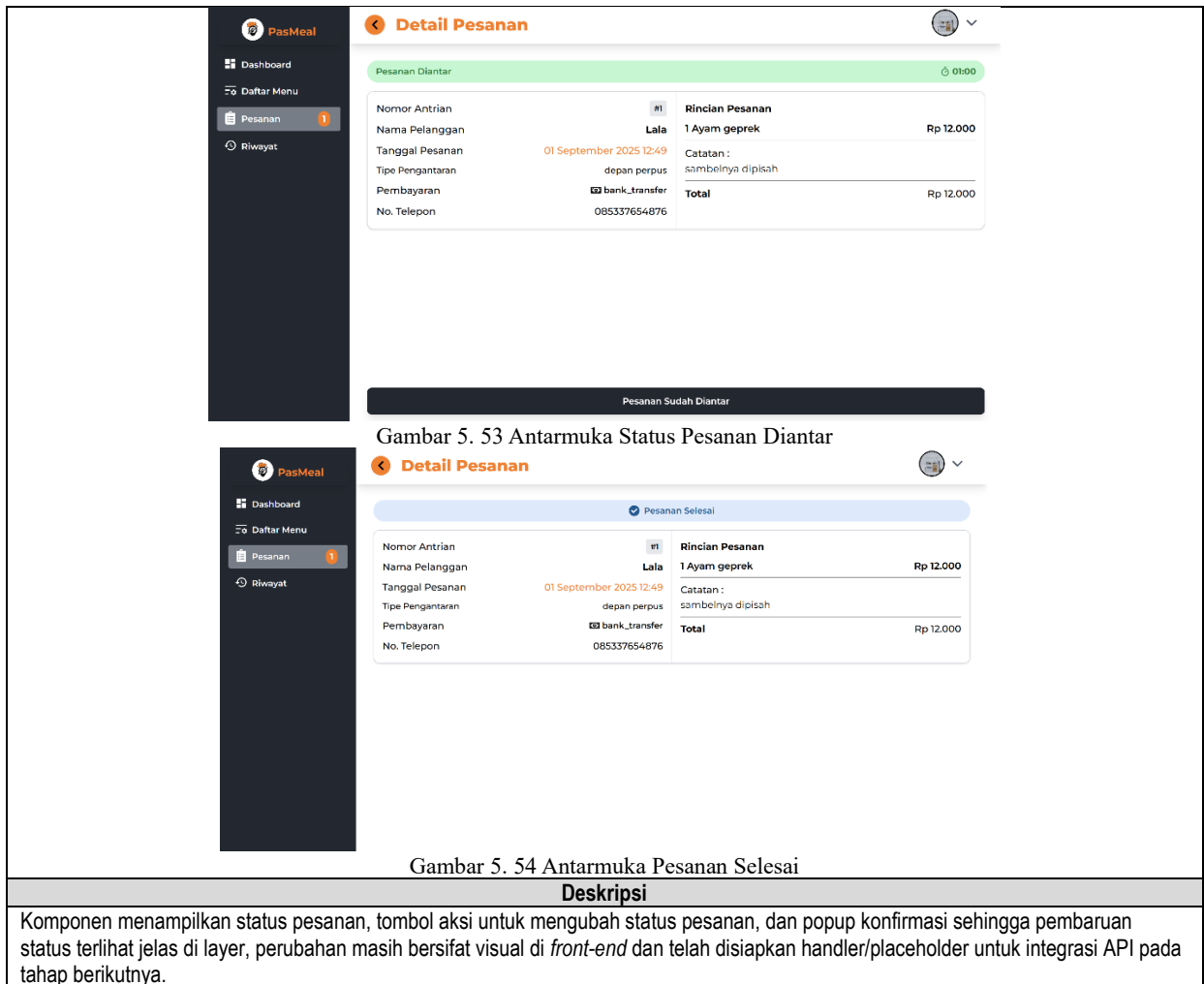
Deskripsi
Komponen Status & Estimasi memuat data dari API untuk merender step status aktif, timer estimasi, dan label tipe pengantaran. Pada bagian Informasi Pelanggan, jika tipe pengantaran Pesan Antar, ditampilkan kolom Diantar ke beserta Nama, No WhatsApp, dan Catatan; jika Ambil Sendiri, kolom diantar ke tidak ditampilkan.

8. Mengubah status pesanan (UI dan Integrasi)

Antarmuka Detail Pesanan untuk mengubah status telah tersedia pada sisi penjual: alur status ditampilkan sesuai tipe pengantaran (Pesan Antar: Menunggu Diproses, Diproses, Pesanan Diantar , Pesanan Selesai; Ambil Sendiri: Menunggu Diproses, Diproses, Siap Diambil , Pesanan Selesai) dengan tombol aksi yang berubah mengikuti tahap, saat ini bersifat UI saja.

Tabel 5. 32 Increment - Mengubah status pesanan (UI dan Integrasi)

Tampilan Status Pesanan	
	Gambar 5. 51 Antarmukan Status Menunggu Diproses
	Gambar 5. 52 Antarmuka Status Pesanan Diproses



9. Mengirim notifikasi pesanan masuk (Integrasi)

Fitur notifikasi pesanan masuk sudah terintegrasi: badge angka pada menu Pesanan di sidebar otomatis menampilkan jumlah order baru dari *back-end*, sehingga penjual langsung tahu ada pesanan masuk tanpa membuka halaman.

Tabel 5. 33 Increment - Mengirim notifikasi pesanan masuk (Integrasi)

Tampilan Badge Notifikasi Pesanan Masuk	
Gambar 5. 55 Integrasi Pesanan Masuk	
Deskripsi	
Komponen Badge Pesanan memuat jumlah pesanan baru dari API saat halaman dibuka dan memperbaruinya secara dinamis). Angka disembunyikan saat bernilai nol dan selalu sinkron dengan data server, sehingga indikator tetap akurat.	

5.1.4.4 Sprint Review Ke-3

Sprint Review ke-3 dilakukan pada akhir iterasi sprint ketiga untuk meninjau serta memverifikasi increment yang telah diselesaikan. Dalam sesi ini, backlog item yang sudah terimplementasi dan terintegrasi ditampilkan dan dievaluasi guna memperoleh masukan. Feedback yang diperoleh kemudian dimanfaatkan sebagai acuan dalam penyempurnaan dan penyusunan rencana pada sprint selanjutnya.

Tabel 5. 34 Sprint Review Ke-3

Kode	Nama Backlog	Status	Hasil dan Umpan Balik
PAS-FE.16	Beranda (Integrasi)	Selesai	Hasil: Fitur ini dinyatakan Selesai karena seluruh fungsi pada halaman beranda, termasuk menampilkan menu terbaru, kios terbaru serta pencarian menu dan kios, telah berhasil diintegrasikan dengan API <i>back-end</i>. Umpan Balik: Fitur berjalan sesuai ekspektasi. Tidak ada perubahan yang diperlukan.
PAS-FE.05	Mengelola keranjang (Integrasi)	Selesai	Hasil: Fitur ini dinyatakan Selesai karena seluruh alur pengelolaan keranjang, termasuk menambahkan, mengubah jumlah, dan menghapus menu, telah berhasil diintegrasikan dengan API <i>back-end</i>. Umpan Balik: Fitur berjalan sesuai ekspektasi. Tidak ada perubahan yang diperlukan.
PAS-FE.06	Membuat pesanan dari keranjang (UI dan Integrasi)	Tidak Selesai	Hasil: Fitur belum selesai karena menu dari keranjang tidak muncul di halaman konfirmasi pesanan, sehingga pesanan tidak bisa dibuat. Umpan Balik: Perlu perbaikan agar menu dari keranjang dapat ditampilkan dengan benar dan pesanan bisa diproses.
PAS-FE.14	Menampilkan tipe pengantaran (Integrasi)	Selesai	Hasil: Fitur ini dinyatakan Selesai karena tipe pengantaran pada pesanan berhasil ditampilkan secara dinamis dengan data dari API <i>back-end</i>. Umpan Balik: Fitur berjalan sesuai ekspektasi. Tidak ada perubahan yang diperlukan.
PAS-FE.07	Menyediakan Payment gateway	Tidak Selesai	Hasil: Fitur Payment Gateway sudah berhasil ditampilkan melalui pop-up, namun alur setelah pengguna menutup pop-up belum berjalan dengan benar. Setelah pop-up ditutup, pengguna tidak dapat melanjutkan proses pemesanan. Umpan Balik: Perlu dilakukan perbaikan pada alur integrasi agar proses pemesanan tetap dapat dilanjutkan meskipun pop-up Payment Gateway

Kode	Nama Backlog	Status	Hasil dan Umpan Balik
			ditutup. Backlog ini akan dijadwalkan ulang untuk diselesaikan pada sprint berikutnya.
PAS-FE.12	Menampilkan daftar pesanan dan detail pesanan (UI dan Integrasi)	Tidak Selesai	Hasil: Tampilan daftar pesanan dan detail pesanan berhasil dibuat serta telah terintegrasi dengan API <i>back-end</i> . Namun, fitur pagination pada daftar pesanan belum diimplementasikan sehingga data pesanan belum dapat ditampilkan secara bertahap. Umpan Balik: Fitur pagination perlu ditambahkan agar daftar pesanan dapat ditampilkan lebih optimal. Backlog ini akan dijadwalkan ulang untuk diselesaikan pada sprint berikutnya.
PAS-FE.08	Menampilkan status dan estimasi waktu pesanan (UI dan Integrasi)	Tidak Selesai	Hasil: Fitur ini belum selesai karena alert untuk pemberitahuan belum ditambahkan, status Menunggu Pesanan belum tersedia, dan integrasi dengan API belum dikerjakan. Umpan Balik: Perlu penambahan alert untuk memberi tahu pembeli bahwa akan ada notifikasi WhatsApp ketika pesanan selesai atau jika pesanan belum selesai tetapi waktu estimasi telah habis. Selain itu, integrasi API perlu dilakukan agar alur status dan estimasi waktu pesanan berjalan dengan benar.
PAS-FE.13	Mengubah status pesanan (UI dan Integrasi)	Tidak Selesai	Hasil: Untuk komponen ubah status pesanan telah berhasil dibuat. Umpan Balik: Integrasi dengan API untuk mengubah status pesanan belum dikerjakan. Oleh karena itu, backlog ini akan dijadwalkan ulang untuk diselesaikan pada sprint berikutnya.
PAS-FE.11	Mengirim notifikasi pesanan masuk (Integrasi)	Selesai	Hasil: Fitur ini dinyatakan Selesai karena notifikasi pesanan masuk berhasil ditampilkan melalui badge pada sidebar dan telah terintegrasi dengan API <i>back-end</i> untuk memperbarui jumlah pesanan baru secara dinamis. Umpan Balik: Fitur berjalan sesuai ekspektasi. Tidak ada perubahan yang diperlukan.

5.1.5 Sprint Ke-4

Sprint ke-4 merupakan kelanjutan proses pengembangan yang dilaksanakan pada periode 22–30 Agustus. Fokusnya adalah perbaikan dan penyempurnaan fungsionalitas yang sudah dibangun, melanjutkan integrasi serta stabilisasi alur pemesanan agar pengalaman pengguna lebih mulus. Aktivitas Scrum, perencanaan, implementasi, dan evaluasi tetap dijalankan untuk memastikan setiap increment memenuhi tujuan dan siap dipresentasikan pada akhir sprint.

5.1.5.1 Sprint Planning Ke-4

Sprint planning ke-4 berfokus pada perbaikan alur pemesanan, penyempurnaan pembayaran, perbaikan daftar dan detail pesanan, integrasi status dan estimasi waktu, integrasi ubah status, serta UI notifikasi pesanan selesai. Tujuannya memastikan rangkaian pembuatan, pembayaran, pemrosesan, dan penyelesaian terhubung ke API *back-end* dan siap dipakai. Aktivitas Scrum tetap dijalankan untuk menjaga arah dan kualitas hasil.

Tabel 5. 35 Sprint Planning Ke-4

Kode	Nama Backlog	Estimasi Waktu	Prioritas
PAS-FE.06	Membuat pesanan dari keranjang (Perbaikan)	7 Hari	Tinggi
PAS-FE.07	Menyediakan Payment gateway (Perbaikan)		Tinggi
PAS-FE.12	Menampilkan daftar pesanan dan detail pesanan (Perbaikan)		Tinggi
PAS-FE.08	Menampilkan status dan estimasi waktu pesanan (UI dan Integrasi)		Tinggi
PAS-FE.13	Mengubah status pesanan (Integrasi)		Tinggi

Kode	Nama Backlog	Estimasi Waktu	Prioritas
PAS-FE.09	Mengirim notifikasi pesanan selesai (UI)		Tinggi

5.1.5.2 Sprint Backlog Ke-4

Sprint backlog pada sprint ke-4 memuat tugas perbaikan dan integrasi API, menyempurnakan pembuatan pesanan dari keranjang, memperbaiki alur payment gateway, merapikan tampilan serta integrasi daftar dan detail pesanan, mengembangkan status dan estimasi waktu pesanan (UI dan integrasi), mengintegrasikan perubahan status pesanan, serta menyiapkan UI notifikasi pesanan selesai. Pekerjaan diatur per halaman dan integrasi agar penyelesaiannya terstruktur dalam waktu sprint.

Tabel 5. 36 Sprint Backlog Ke-4

Kode	Nama Backlog	Kebutuhan	Waktu Terpakai
PAS-FE.06	Membuat pesanan dari keranjang (Perbaikan)	Menampilkan menu dari keranjang di halaman konfirmasi pesanan.	7 hari
PAS-FE.07	Menyediakan Payment gateway (Perbaikan)	Memperbaiki alur agar pemesanan tetap bisa dilanjutkan jika menutup pop-up.	
PAS-FE.12	Menampilkan daftar pesanan dan detail pesanan (Perbaikan)	Menambahkan fitur pagination pada daftar pesanan agar data ditampilkan secara bertahap.	
PAS-FE.08	Menampilkan status dan estimasi waktu pesanan (UI dan Integrasi)	Menambahkan status Menunggu Pesanan agar alur status lebih lengkap.	
		Menambahkan alert untuk memberitahu pembeli tentang notifikasi WhatsApp ketika pesanan selesai atau jika waktu estimasi habis.	
		Mengintegrasikan API agar estimasi waktu pesanan ditampilkan dengan benar.	
PAS-FE.13	Mengubah status pesanan (UI dan Integrasi)	Mengintegrasikan halaman konfirmasi pesanan dengan	
		Midtrans Snap untuk memproses pembayaran.	
PAS-FE.12	Menampilkan daftar pesanan dan detail pesanan (UI dan Integrasi)	Halaman daftar pesanan.	
		Halaman detail pesanan.	
		Integrasi API untuk menampilkan daftar pesanan dan detail pesanan.	
PAS-FE.08	Menampilkan status dan estimasi waktu pesanan (UI dan Integrasi)	Halaman status pesanan.	
		Integrasi API status pesanan dan estimasi waktu pesanan.	
PAS-FE.13	Mengubah status pesanan (Integrasi)	Mengintegrasikan API agar status pesanan dapat diubah dan tersimpan dengan benar.	
PAS-FE.09	Mengirim notifikasi pesanan selesai (UI)	Menampilkan status pesanan Selesai di halaman pembeli ketika pesanan telah selesai.	

5.1.5.3 Increment Ke-4

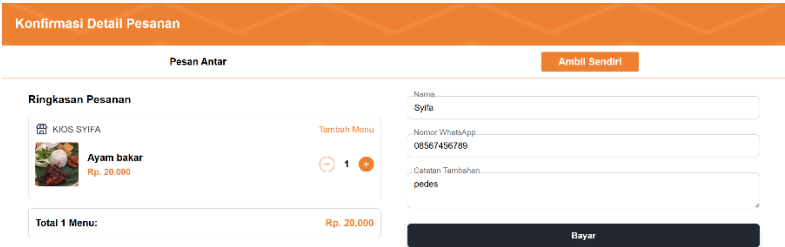
Pada akhir Sprint 4, increment yang dihasilkan adalah aplikasi berfungsi penuh dengan seluruh antarmuka terintegrasi ke API *back-end*. Cakupan pekerjaan meliputi perbaikan pembuatan pesanan dari keranjang, perbaikan integrasi *payment gateway*, perbaikan daftar dan detail pesanan, penambahan status serta estimasi waktu pesanan (UI dan integrasi), integrasi ubah status pesanan, serta UI pesanan selesai.

1. Membuat pesanan dari keranjang (Perbaikan)

Fitur Membuat Pesanan dari Keranjang (Perbaikan) menghadirkan halaman Konfirmasi Detail Pesanan tempat pengguna meninjau item keranjang, mengatur jumlah, memilih tipe pengantaran (Pesanan

Antar / Ambil Sendiri), melengkapi data pemesan, lalu menekan Bayar untuk membuat pesanan dan melanjutkan ke pembayaran.

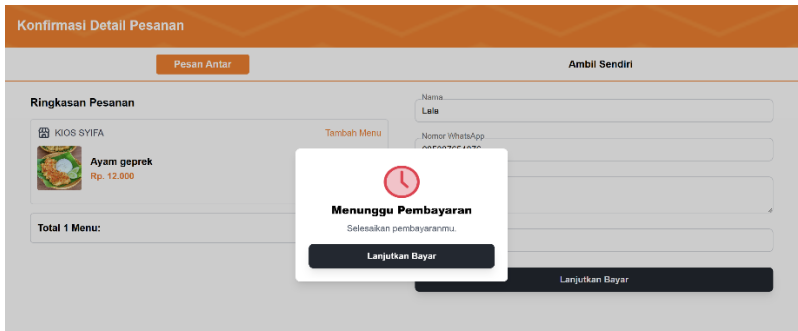
Tabel 5. 37 Increment - Membuat pesanan dari keranjang (Perbaikan)

Tampilan Konfirmasi Detail Pesanan (Perbaikan)	
	
Gambar 5. 56 Integrasi Halaman Konfirmasi Detail Pesanan	
Deskripsi	
Ringkasan pesanan kini terhubung penuh dengan data keranjang, setiap item menampilkan foto, nama, harga, serta kontrol tambah/kurang yang memperbarui total secara otomatis. Tersedia tombol Tambah Menu, di bagian atas tersedia pilihan tipe pengantaran (Pesanan Antar/Ambil Sendiri), dan formulir Nama, Nomor WhatsApp, serta Catatan, jika memilih Pesanan Antar, muncul kolom Diantar ke. Tombol Bayar hanya aktif ketika data wajib telah terisi.	

2. Menyediakan *Payment gateway* (Perbaikan)

Fitur *Payment Gateway* memungkinkan pengguna melanjutkan pembayaran dari halaman Konfirmasi Detail Pesanan. Setelah menekan Lanjutkan Bayar, aplikasi memanggil API untuk mengambil token transaksi dan menampilkan *pop-up* pembayaran; bila pembayaran tertunda, muncul dialog “Menunggu Pembayaran” dengan tombol Lanjutkan Bayar untuk membuka kembali *pop-up*.

Tabel 5. 38 Increment - Menyediakan *Payment gateway* (Perbaikan)

Tampilan <i>Payment gateway</i>	
	
Gambar 5. 57 Integrasi <i>Payment Gateway</i>	
Deskripsi	
Saat pengguna menutup <i>pop-up</i>, aplikasi menyimpan konteks transaksi dan menampilkan dialog “Menunggu Pembayaran”. Menekan Lanjutkan Bayar akan memanggil ulang alur pembayaran dan membuka kembali modal gateway, sehingga proses dapat diteruskan mulus tanpa perlu mengisi ulang pesanan	

3. Menampilkan daftar pesanan dan detail pesanan (Perbaikan)

Fitur Daftar Pesanan & Detail Pesanan kini mendukung pagination. Penjual dapat menelusuri pesanan dalam halaman-halaman dengan kontrol sebelumnya/berikutnya dan nomor halaman, lalu membuka Detail untuk melihat informasi lengkap tiap pesanan.

Tabel 5. 39 Increment - Menampilkan daftar pesanan dan detail pesanan (Perbaikan)

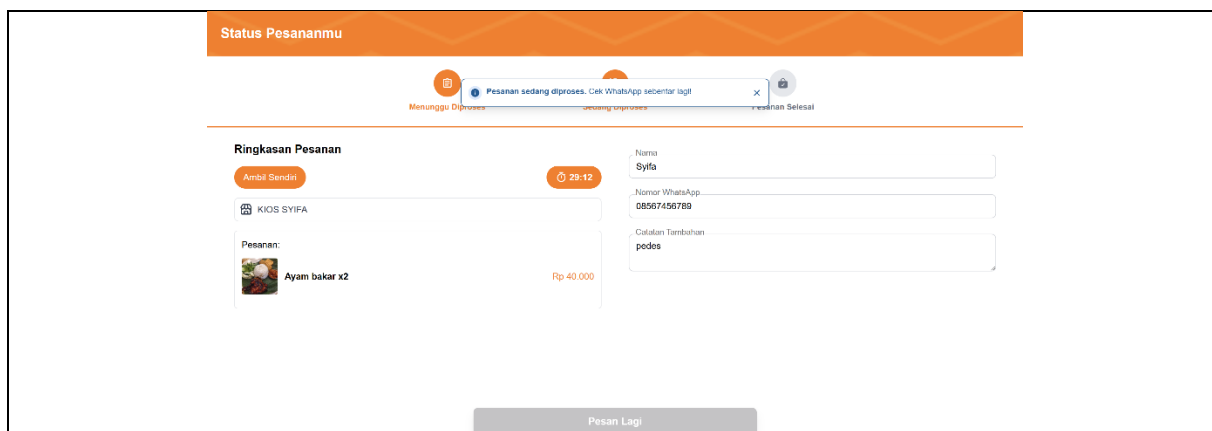
Tampilan Daftar Pesanan dan Detail Pesanan (Perbaikan)																							
PasMeal Dashboard Daftar Menu Pesanan2 Riwayat Pesanan <table><tr><th>No</th><th>Nama Pelanggan</th><th>Tanggal</th><th>Harga</th><th>Status Pesanan</th><th>Detail</th></tr><tr><td>#1</td><td>SINTA</td><td>01 September 2025 13:43</td><td>Rp. 20.000</td><td>Menunggu Diproses</td><td>Detail</td></tr><tr><td>#2</td><td>LILY</td><td>01 September 2025 13:44</td><td>Rp. 20.000</td><td>Menunggu Diproses</td><td>Detail</td></tr></table> <1>						No	Nama Pelanggan	Tanggal	Harga	Status Pesanan	Detail	#1	SINTA	01 September 2025 13:43	Rp. 20.000	Menunggu Diproses	Detail	#2	LILY	01 September 2025 13:44	Rp. 20.000	Menunggu Diproses	Detail
No	Nama Pelanggan	Tanggal	Harga	Status Pesanan	Detail																		
#1	SINTA	01 September 2025 13:43	Rp. 20.000	Menunggu Diproses	Detail																		
#2	LILY	01 September 2025 13:44	Rp. 20.000	Menunggu Diproses	Detail																		
Gambar 5. 58 Antarmuka dan Integrasi Halaman Daftar Peasanan - Pagination																							
Deskripsi																							
Komponen mengambil data pesanan per halaman dari <i>back-end</i> dan merender tabel berisi No, Nama Pelanggan, Tanggal, Harga, dan Status, dengan tombol Detail di tiap baris. Kontrol pagination mengirim query page ke API, memperbarui tampilan tanpa memuat ulang seluruh halaman, dan menjaga kinerja ketika jumlah pesanan besar.																							

4. Menampilkan status dan estimasi waktu pesanan (UI dan Integrasi)

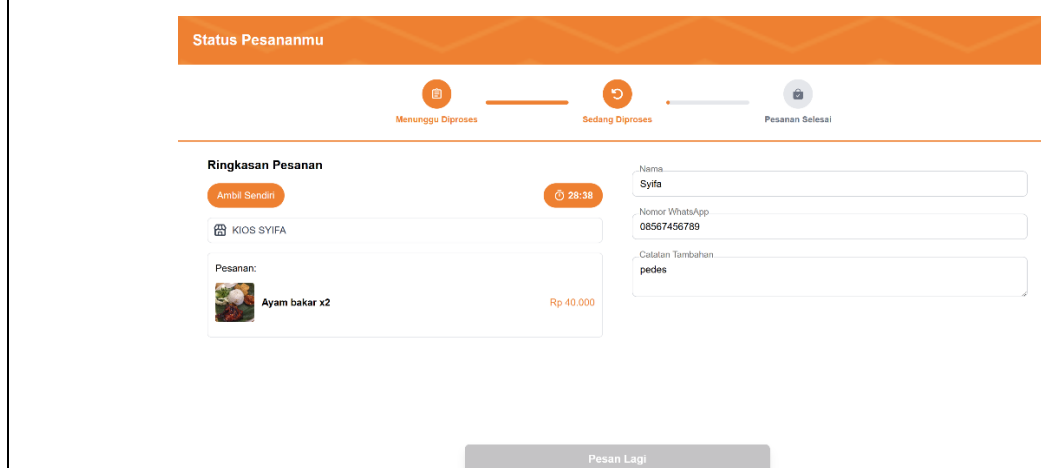
Halaman Status Pesananmu menampilkan progres pesanan secara real-time dari Menunggu Diproses, Sedang Diproses, Pesanan Selesai, lengkap dengan penghitung mundur estimasi dan tipe pengantaran (Ambil Sendiri/Pesan Antar). Jika Pesan Antar, kolom Diantar ke ditampilkan. Saat status berubah atau estimasi terlampaui, muncul notifikasi di bagian atas.

Tabel 5. 40 Increment - Menampilkan status dan estimasi waktu pesanan (UI dan Integrasi)

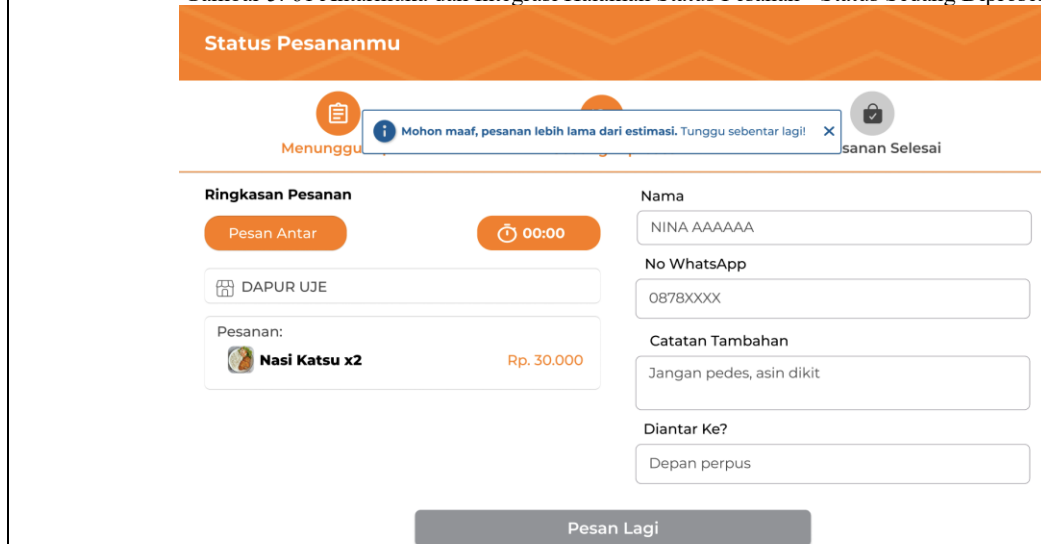
Tampilan Status Pesanan	
Status Pesananmu Menunggu Diproses Sedang Diproses Pesanan Selesai Ringkasan Pesanan Ambil Sendiri Menunggu KIOS SYIFA Pesanan: Ayam bakar x2Rp 40.000 Nama: Syifa Nomor WhatsApp: 08567456789 Catatan Tambahan: pedes Pesan Lagi	
Gambar 5. 59 Antarmuka dan Integrasi Halaman Status Pesanan - Status Menunggu Diproses	



Gambar 5. 60 Antarmuka Halaman Status Pesanan - Alert Pemberitahuan



Gambar 5. 61 Antarmuka dan Integrasi Halaman Status Pesanan - Status Sedang Diproses



Gambar 5. 62 Antarmuka Halaman Status Pesanan - Alert Pesanan Terlambat

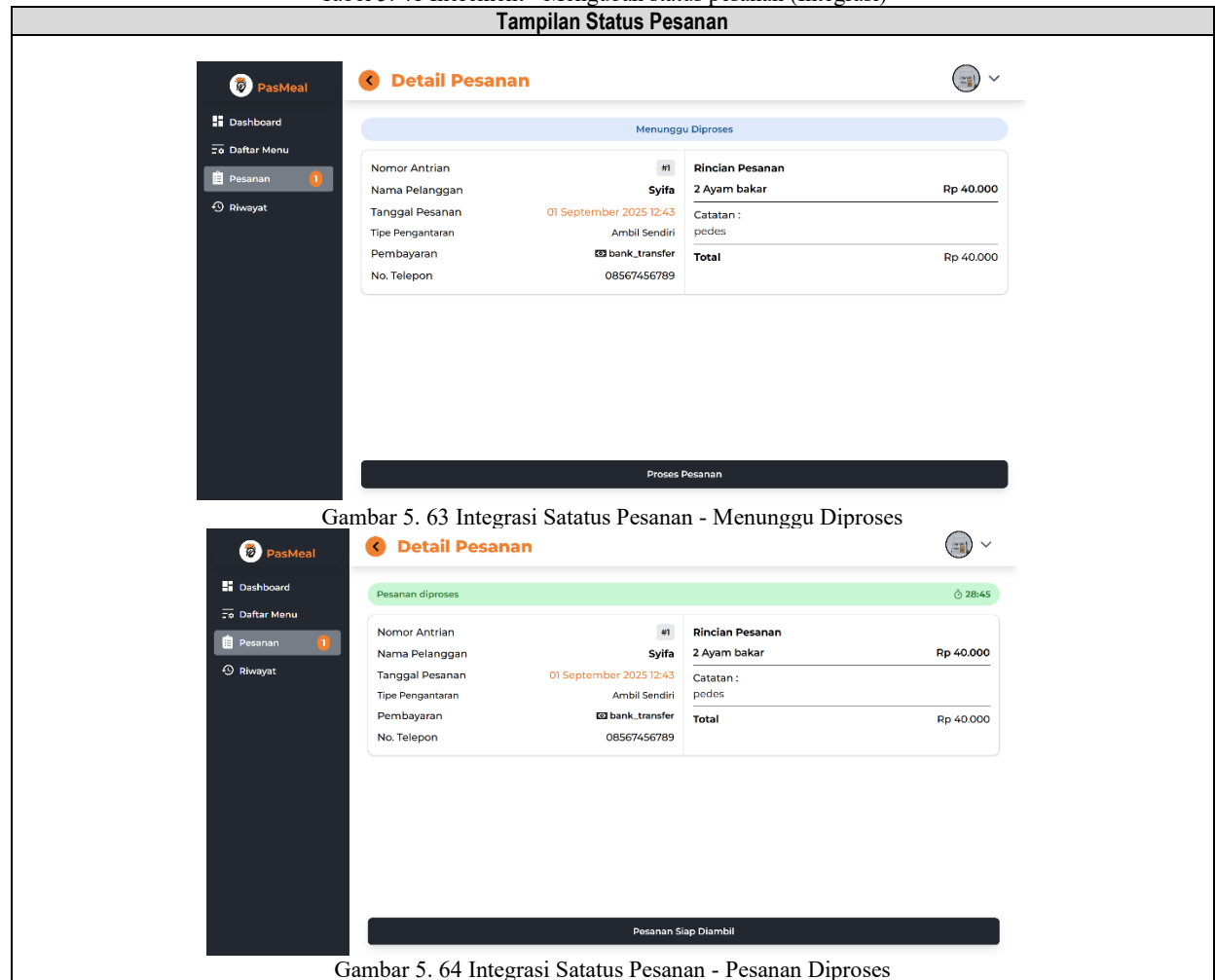
Deskripsi

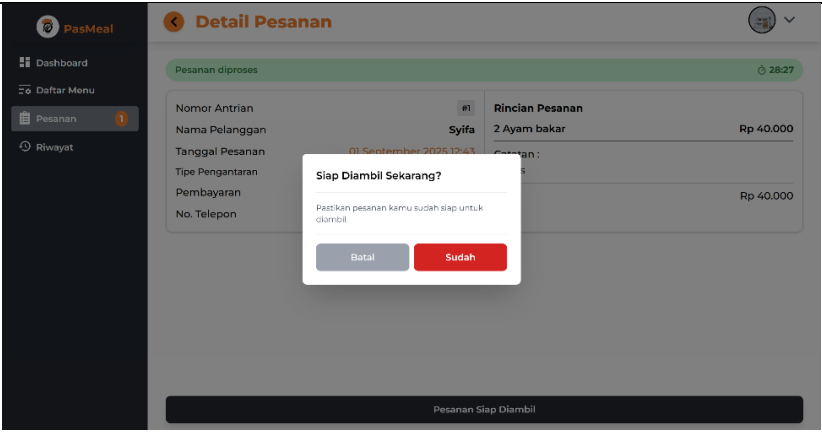
Komponen Status & Estimasi mengambil data dari API untuk menampilkan tahap status pesanan (Menunggu, Diproses, Selesai), timer estimasi yang menghitung mundur, dan tipe pengantaran. Jika pembeli memilih Pesan Antar, kolom Diantar ke ikut muncul. Saat status berubah atau waktu estimasi habis, sistem memunculkan notifikasi singkat, dan tampilan otomatis diperbarui agar sesuai dengan data dari API *back-end*.

5. Mengubah status pesanan (Integrasi)

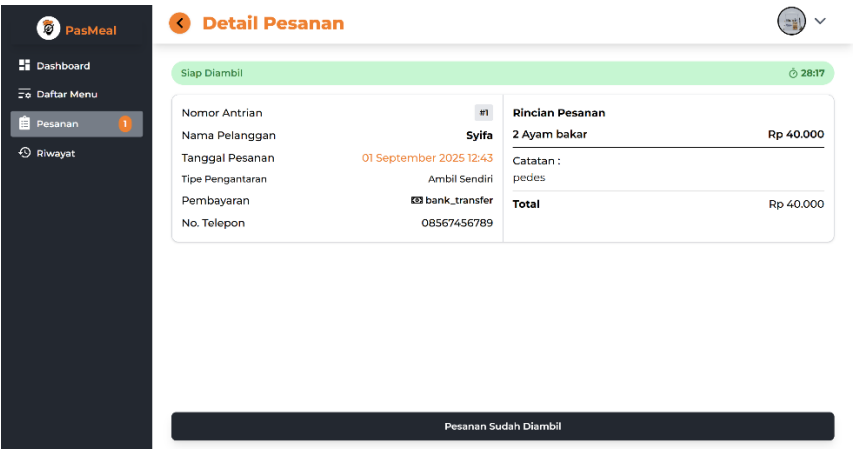
Fitur Mengubah Status Pesanan (UI & Integrasi) memungkinkan penjual memperbarui status langsung dari halaman Detail Pesanan dengan tombol aksi yang menyesuaikan tipe pengantaran. Alur Pesan Antar: Menunggu Diproses, Diproses, Diantar, Sudah Diantar. Alur Ambil Sendiri: Menunggu Diproses, Diproses, Siap Diambil, Sudah Diambil. Setiap kali tombol ditekan, status tampilannya berubah dan tersinkron ke API *back-end*.

Tabel 5. 41 Increment - Mengubah status pesanan (Integrasi)

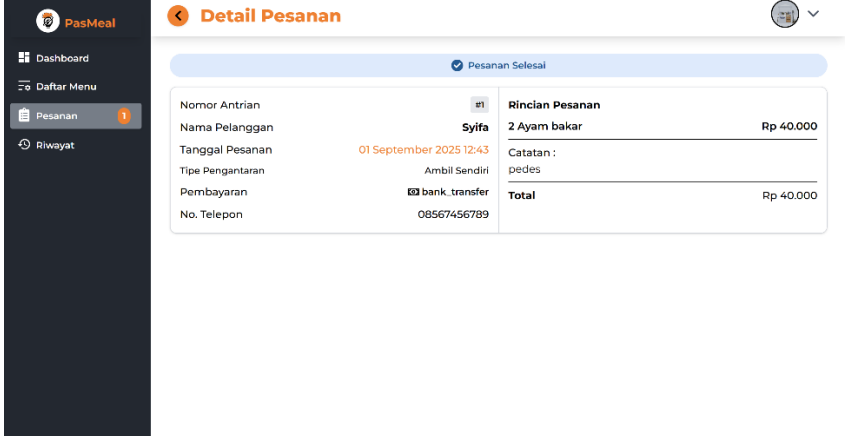




Gambar 5. 65 Integrasi Satatus Pesanan - Konfirmasi Tipe Pengantaran



Gambar 5. 66 Integrasi Satatus Pesanan - Siap Diambil



Gambar 5. 67 Integrasi Satatus Pesanan - Pesanan selesai

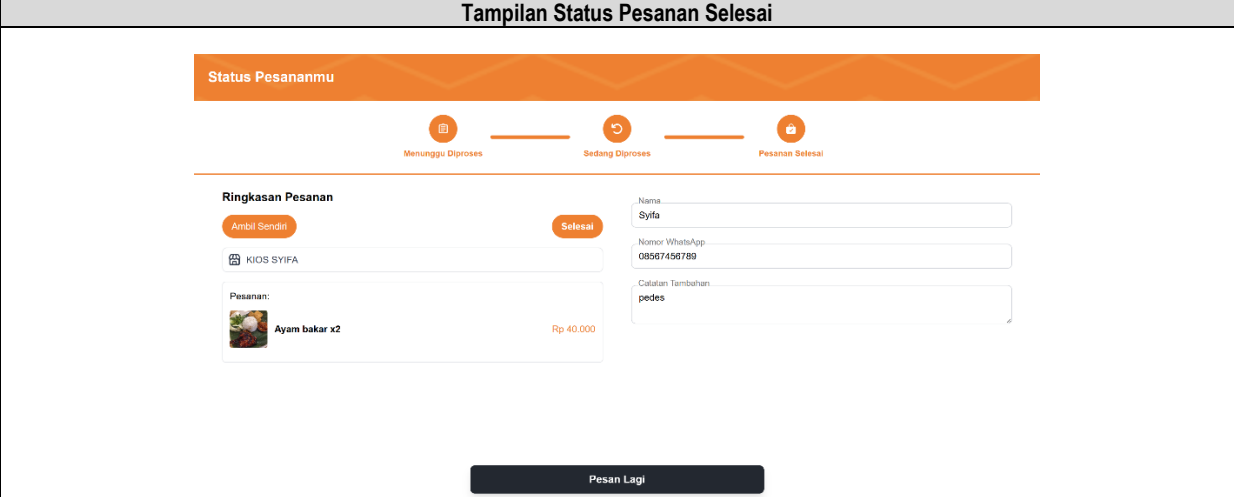
Deskripsi

Komponen ini menampilkan status pesanan beserta tombol aksi yang otomatis menyesuaikan tipe pengantaran; saat tombol ditekan, aplikasi mengirim permintaan ubah status ke API lalu memperbarui tampilan di Detail Pesanan dan Daftar Pesanan. Untuk Pesan Antar, urutan tombol berganti dari Proses Pesanan, Kirim Pesanan, Tandai Sudah Diantar, sedangkan untuk Ambil Sendiri berganti dari Proses Pesanan, Tandai Siap Diambil, Tandai Sudah Diambil, sehingga penjual dapat mengonfirmasi progres pesanan dengan jelas tanpa berpindah halaman.

6. Mengirim notifikasi pesanan selesai (UI)

Fitur Mengirim Notifikasi Pesanan Selesai menampilkan pemberitahuan visual saat pesanan berstatus Selesai pada halaman Status Pesananmu. Indikator tahap berpindah ke Pesanan Selesai dan label Selesai muncul di ringkasan, sehingga pengguna tahu pesanan siap diambil/diterima.

Tabel 5. 42 Increment - Mengirim notifikasi pesanan selesai (UI)

Tampilan Status Pesanan Selesai	
	
Deskripsi	
Komponen UI merender status akhir Selesai pada status pesanan, menampilkan label Selesai di ringkasan pesanan, serta mempertahankan informasi pesanan (kios, item, total, dan data pelanggan). Pemberitahuan ini bersifat tampilan di <i>front-end</i>, pengiriman notifikasi dikelola oleh <i>back-end</i>, dan UI hanya merefleksikan perubahan status tersebut.	

5.1.5.4 Sprint Review Ke-4

Sprint Review ke-4 dilakukan pada akhir iterasi sprint keempat untuk meninjau serta memverifikasi increment yang telah diselesaikan. Dalam sesi ini, backlog item yang sudah terimplementasi dan terintegrasi ditampilkan dan dievaluasi guna memperoleh masukan. Umpan balik yang diperoleh kemudian dimanfaatkan sebagai acuan dalam penyempurnaan dan penyusunan rencana pada tahap berikutnya.

Tabel 5. 43 Sprint Review Ke-4

Kode	Nama Backlog	Status	Hasil dan Umpan Balik
PAS-FE.06	Membuat pesanan dari keranjang (Perbaikan)	Selesai	Hasil: Fitur ini dinyatakan selesai karena menu dari keranjang sudah berhasil ditampilkan di halaman konfirmasi pesanan dan pesanan dapat dibuat dengan benar. Umpan Balik: Fitur berjalan sesuai ekspektasi. Tidak ada perubahan yang diperlukan.
PAS-FE.07	Menyediakan Payment gateway (Perbaikan)	Selesai	Hasil: Fitur ini dinyatakan selesai karena pop-up Payment Gateway berhasil ditampilkan dan alur pemesanan dapat dilanjutkan setelah pop-up ditutup. Umpan Balik: Fitur berjalan sesuai ekspektasi. Tidak ada perubahan yang diperlukan.
PAS-FE.12	Menampilkan daftar pesanan dan detail pesanan (Perbaikan)	Selesai	Hasil: Fitur ini dinyatakan selesai karena daftar pesanan, detail pesanan, dan pagination telah berhasil dibuat dan terintegrasi dengan API <i>back-end</i>. Umpan Balik: Fitur berjalan sesuai ekspektasi. Tidak ada perubahan yang diperlukan.

Kode	Nama Backlog	Status	Hasil dan Umpan Balik
PAS-FE.08	Menampilkan status dan estimasi waktu pesanan (UI dan Integrasi)	Selesai	Hasil: Fitur ini dinyatakan selesai karena status Menunggu Pesanan, estimasi waktu, dan alert pemberitahuan telah berhasil ditambahkan, serta integrasi dengan API <i>back-end</i> berjalan dengan baik. Umpan Balik: Fitur berjalan sesuai ekspektasi. Tidak ada perubahan yang diperlukan.
PAS-FE.13	Mengubah status pesanan (UI dan Integrasi)	Selesai	Hasil: Fitur ini dinyatakan selesai karena status pesanan dapat diubah dengan benar melalui antarmuka pengguna dan integrasi dengan API <i>back-end</i> telah berhasil. Umpan Balik: Fitur berjalan sesuai ekspektasi. Tidak ada perubahan yang diperlukan.
PAS-FE.09	Mengirim notifikasi pesanan selesai (UI)	Selesai	Hasil: Fitur ini dinyatakan selesai karena status pesanan berhasil ditampilkan sebagai Selesai di halaman status pesanan. Umpan Balik: Fitur berjalan sesuai ekspektasi. Tidak ada perubahan yang diperlukan.

5.2 Delivery

Sesudah seluruh tahapan Scrum Process dilaksanakan, langkah berikutnya adalah Delivery. Pada tahap ini, produk yang sudah berfungsi dengan baik, telah melewati proses pengujian, serta sesuai dengan kebutuhan pengguna diserahkan kepada pemilik produk. Dalam konteks pengembangan *front-end*, kegiatan delivery dilakukan dengan memberikan akses terhadap aplikasi pemesanan makanan online di Kantin Universitas Pasundan Setiabudi yang telah berhasil di-*deploy* pada layanan hosting.

Hasil dari tahap ini berupa URL aplikasi *front-end* yang dapat langsung digunakan oleh pengguna, disertai dengan dokumentasi fitur utama yang sudah selesai dikembangkan. Proses penyerahan ini menjadi tanda bahwa pengembangan antarmuka pengguna beserta integrasinya dengan API *back-end* telah selesai.

5.2.1 Pengujian Perangkat Lunak

Pengujian perangkat lunak dilakukan untuk memastikan bahwa aplikasi yang dibangun berfungsi sebagaimana mestinya dan sesuai dengan kebutuhan pengguna. Tahap ini juga membantu dalam meningkatkan kualitas perangkat lunak, baik dari sisi fitur yang sudah berjalan dengan benar maupun dari temuan kesalahan yang perlu diperbaiki.

Salah satu pendekatan yang digunakan adalah black box testing. Metode ini berorientasi pada pengujian fungsionalitas sistem tanpa melihat detail kode program di dalamnya. Penguji cukup memberikan input tertentu dan kemudian memvalidasi apakah output yang dihasilkan sudah sesuai dengan yang diharapkan. Hasil dari proses pengujian tersebut disajikan pada Tabel berikut.

Tabel 5. 44 Pengujian Autentikasi

Kode Test Skenario	Test Skenario	Kode Test Case	Test Case	Test Data	Expested Result	Actual Result	Pass/Fail
TS-01	Cek Fungsi Register	TC1-01	Register Akun dengan data valid	Nama: "Syifa" Nomor WhatsApp: "086789456345"	Pengguna otomatis diarahkan kehalaman Registrasi Toko.	Pengguna otomatis diarahkan kehalaman Registrasi Toko.	Pass

Kode Test Skenario	Test Skenario	Kode Test Case	Test Case	Test Data	Expeded Result	Actual Result	Pass/Fail
				Email: "syifa@gmail.com" Kata Sandi : "syifa123" Konfirmasi Kata Sandi: "syifa123"			
		TC1-02	Register Akun dengan nomor WhatsApp yang sudah terdaftar	Nama: "Syifa" Nomor WhatsApp: "086789456345" Email: "syifa@gmail.com" Kata Sandi : "syifa321" Konfirmasi Kata Sandi: "syifa321"	Pesan error di atas form, seperti "Nomor WhatsApp sudah digunakan".	Pesan error di atas form, seperti "Nomor WhatsApp sudah digunakan".	Pass
		TC1-03	Register Akun dengan Kata Sandi kurang dari 8 karakter	Nama: "Syifa" Nomor WhatsApp: "086789456345" Email: "syifa@gmail.com" Kata Sandi : "syifa" Konfirmasi Kata Sandi: "syifa"	Pesan error di bawah kolom kata sandi, seperti "Kata sandi minimal 8 karakter".	Pesan error di bawah kolom kata sandi, seperti "Kata sandi minimal 8 karakter".	Pass
		TC1-04	Register Akun dengan Kata Sandi dan Konfirmasi Kata Sandi tidak sama	Nama: "Syifa" Nomor WhatsApp: "086789456345" Email: "syifa@gmail.com" Kata Sandi : "syifa123" Konfirmasi Kata Sandi: "syifa"	Pesan error di bawah kolom konfirmasi kata sandi, seperti "Konfirmasi tidak sama dengan kata sandi".	Pesan error di bawah kolom konfirmasi kata sandi, seperti "Konfirmasi tidak sama dengan kata sandi".	Pass
		TC1-05	Register Akun dengan form kosong	Nama: "" Nomor WhatsApp: "" Email: "" Kata Sandi : "" Konfirmasi Kata Sandi: ""	Tombol Lanjut disable.	Tombol Lanjut disable.	Pass
TS-02	Cek Fungsi Register Toko	TC2-01	Register Toko dengan data valid	Nama Toko: "Kios Syifa" Nama Bank: "BNI" Nomor Rekening: "9645731256"	Pengguna otomatis diarahkan kehalaman Verifikasi OTP.	Pengguna otomatis diarahkan kehalaman Verifikasi OTP.	Pass
		TC2-02	Register Toko dengan nomor Nama Toko yang sudah terdaftar	Nama Toko: "Kios Syifa" Nama Bank: "BNI" Nomor Rekening: "9645731256"	Pesan error diatas form "Nama toko sudah digunakan, silakan pilih nama lain".	Pesan error diatas form "Nama toko sudah digunakan, silakan pilih nama lain".	Pass
		TC2-03	Register Toko dengan Nomor Rekening menggunakan huruf	Nama Toko: "Kios Syifa" Nama Bank: "BNI" Nomor Rekening: "m0208s14073"	Pesan error di bawah kolom Nomor Rekening, seperti "Nomor rekening harus berupa angka".	Pesan error di bawah kolom Nomor Rekening, seperti "Nomor rekening harus berupa angka".	Pass
		TC2-04	Register Toko dengan form kosong	Nama Toko: "" Nama Bank: "" Nomor Rekening: ""	Tombol Lanjut disable.	Tombol Lanjut disable.	Pass

Kode Test Skenario	Test Skenario	Kode Test Case	Test Case	Test Data	Expeded Result	Actual Result	Pass/Fail
TS-03	Cek Fungsi Verifikasi Akun Dengan OTP	TC3-01	Verifikasi Akun dengan OTP yang valid	OTP "kode-otp-valid"	Notifikasi berhasil "Selamat Akun Anda telah terdaftar".	Notifikasi berhasil "Selamat Akun Anda telah terdaftar".	Pass
		TC3-02	Verifikasi Akun dengan OTP yang Salah	OTP: "222222"	Notifikasi error "OTP salah, kadaluarsa, atau sudah digunaka".	Notifikasi error "OTP salah, kadaluarsa, atau sudah digunaka".	Pass
		TC3-03	Meminta kirimn ulang OTP	-	Pesan konfirmasi "Kode OTP baru telah dikirim".	Pesan konfirmasi "Kode OTP baru telah dikirim".	Pass
TS-04	Cek Fungsi Login	TC4-01	Login dengan data yg benar	Nama: "Syifa" Kata Sandi : "syifa123"	Pengguna diarahkan kehalaman dashboard.	Pengguna diarahkan kehalaman dashboard.	Pass
		TC4-02	Login dengan email/kata sandi yang salah	Nama: "Syufa" Kata Sandi : "syifa122"	Notifikasi error "Nama atau kata sandi salah".	Notifikasi error "Nama atau kata sandi salah".	Pass
		TC4-03	Login dengan form kosong	Nama: "" Kata Sandi : ""	Tombol Masuk disable.	Tombol Masuk disable.	Pass
TS-05	Cek Fungsi Lupa Kata Sandi	TC5-01	Request link untuk reset Kata Sandi dengan Nomor WhatsApp	Nomor WhatsApp: "086789456345"	Pesan konfirmasi "Link reset password telah dikirim ke WhatsApp Anda".	Pesan konfirmasi "Link reset password telah dikirim ke WhatsApp Anda".	Pass
		TC5-02	Reset Kata Sandi dengan data valid	Kata Sandi Baru: "syifa0208" Konfirmasi Kata Sandi Baru: "syifa0208"	Notifikasi berhasil "Kata sandi telah diperbarui"	Notifikasi berhasil "Kata sandi telah diperbarui"	Pass
		TC5-03	Reset Kata Sandi dengan Kata Sandi Baru dan Konfirmasi Kata Sandi Baru tidak sama	Kata Sandi Baru: "syifa0208" Konfirmasi Kata Sandi Baru: "syifa0207"	Pesan error di bawah kolom konfirmasi kata sandi, seperti "Konfirmasi tidak sama dengan kata sandi baru".	Pesan error di bawah kolom konfirmasi kata sandi, seperti "Konfirmasi tidak sama dengan kata sandi baru".	Pass
		TC5-04	Reset Kata Sandi dengan Kata Sandi Baru kurang dari 8 karakter	Kata Sandi Baru: "syifa" Konfirmasi Kata Sandi Baru: "syifa"	Pesan error "Kata sandi minimal 8 karakter".	Pesan error "Kata sandi minimal 8 karakter".	Pass
		TC5-05	Request link untuk reset Kata Sandi dengan Nomor WhatsApp tidak terdaftar	Nomor WhatsApp: "086789614085"	Pesan error "Nomor WhatsApp tidak ditemukan".	Pesan error "Nomor WhatsApp tidak ditemukan".	Pass
		TC5-06	Resat Kata Sandi dengan form kosong	Kata Sandi Baru: "" Konfirmasi Kata Sandi Baru: ""	1. Pesan error di bawah kolom Kata Sandi Baru, seperti "Kata sandi wajib Baru diisi".	1. Pesan error di bawah kolom Kata Sandi Baru, seperti "Kata sandi wajib Baru diisi".	Pass

Kode Test Skenario	Test Skenario	Kode Test Case	Test Case	Test Data	Expeded Result	Actual Result	Pass/Fail
					2. Pesan error di bawah kolom Konfirmasi Kata Sandi Baru, seperti "Konfirmasi kata sandi wajib Baru diisi".	2. Pesan error di bawah kolom Konfirmasi Kata Sandi Baru, seperti "Konfirmasi kata sandi wajib Baru diisi".	
TS-06	Cek Fungsi Melengkapi dan Mengubah Profil	TC6-01	Mengubah data penjual	Nama: "Syifa" Nomor WhatsApp: "086789456345" Email: "syifa123@gmail.com"	Notifikasi berhasil "Profil berhasil diperbarui".	Notifikasi berhasil "Profil berhasil diperbarui".	Pass
		TC6-02	Melengkapi data kios	Gambar: "kios.png" Deskripsi Toko: Nasi Ayam	Notifikasi berhasil "Profil toko berhasil diperbarui".	Notifikasi berhasil "Profil toko berhasil diperbarui".	Pass
TS-07	Cek Fungsi Logout	TC7-01	Klik tombol Keluar		Pengguna berhasil logout dan diarahkan ke halaman login	Pengguna berhasil logout dan diarahkan ke halaman login	Pass

Tabel 5. 45 Pengujian Menampilkan Kios dan Menu

Kode Test Skenario	Test Skenario	Kode Test Case	Test Case	Test Data	Expeded Result	Actual Result	Pass/Fail
TS-08	Cek Fungsi Kios	TC08-01	Menampilkan semua kios Ketika halaman dimuat		Menampilkan seluruh kios	Menampilkan seluruh kios	Pass
		TC08-02	Cari kios	Ketik "Kios Syifa"	Menampilkan kios dengan nama "Kios Syifa"	Menampilkan kios dengan nama "Kios Syifa"	Pass
		TC08-03	Menampilkan menu	Klik "Lihat Menu" pada salah satu kios	Menampilkan semua menu pada kios yang dipilih	Menampilkan semua menu pada kios yang dipilih	Pass
		TC08-02	Cari kios	Ketik "Kios Syifa"	Menampilkan kios dengan nama "Kios Syifa"	Menampilkan kios dengan nama "Kios Syifa"	Pass
		TC08-03	Menampilkan menu	Klik "Lihat Menu" pada salah satu kios	Menampilkan semua menu pada kios yang dipilih	Menampilkan semua menu pada kios yang dipilih	Pass
		TC08-04	Cari menu	Ketik "Ayam"	Sistem menampilkan daftar menu yang mengandung kata "Ayam" (contoh: Ayam geprek, Ayam bakar)	Sistem menampilkan daftar menu yang mengandung kata "Ayam" (contoh: Ayam geprek, Ayam bakar)	Pass
		TC08-05	Menampilkan detail menu	Klik salah satu menu	Menampilkan detail menu yang dipilih	Menampilkan detail menu yang dipilih	Pass

Tabel 5. 46 Pengujian Mengelola Keranjang

Kode Test Skenario	Test Skenario	Kode Test Case	Test Case	Test Data	Expeded Result	Actual Result	Pass/Fail
TS-09	Cek Fungsi Mengelola Keranjang	TC09-01	Menampilkan data keranjang		Halaman menampilkan daftar semua item yang telah ditambahkan	Halaman menampilkan daftar semua item yang telah ditambahkan	Pass

Kode Test Skenario	Test Skenario	Kode Test Case	Test Case	Test Data	Expeded Result	Actual Result	Pass/Fail
		TC09-02	Menambah produk ke keranjang	Menekan tombol "Tambah Keranjang" pada card produk	Badge pada tombol keranjang bertambah +1 sesuai jumlah item dalam keranjang	Badge pada tombol keranjang bertambah +1 sesuai jumlah item dalam keranjang	Pass
		TC09-03	Mengubah kuantitas menu	Pada salah satu menu, klik tombol + untuk menambah kuantitas dari 1 menjadi 2.	1. Angka kuantitas untuk menu menjadi 2. 2. Total keseluruhan berubah secara otomatis.	1. Angka kuantitas untuk menu menjadi 2. 2. Total keseluruhan berubah secara otomatis.	Pass
		TC09-04	Mengubah kuantitas menjadi nol atau kurang	Pada menu dengan kuantitas 1, klik tombol -.	Menu otomatis terhapus dari daftar keranjang	Menu otomatis terhapus dari daftar keranjang	Pass

Tabel 5. 47 Pengujian Membuat Pesanan

Kode Test Skenario	Test Skenario	Kode Test Case	Test Case	Test Data	Expeded Result	Actual Result	Pass/Fail
TS10	Cek Fungsi Konfirmasi Pesanan	TC10-01	Menampilkan data ringkasan pesanan		Halaman menampilkan daftar semua pesanan yang telah dipesan	Halaman menampilkan daftar semua pesanan yang telah dipesan	Pass
		TC10-02	Mengubah kuantitas menu dari halaman konfirmasi	Pada salah satu menu, klik tombol + untuk menambah kuantitas dari 1 menjadi 2.	1. Angka kuantitas untuk menu menjadi 2. 2. Total keseluruhan berubah secara otomatis.	1. Angka kuantitas untuk menu menjadi 2. 2. Total keseluruhan berubah secara otomatis.	Pass
		TC10-03	Mengubah kuantitas menjadi nol atau kurang	Pada menu dengan kuantitas 1, klik tombol -.	Menu otomatis terhapus dari ringkasan pesanan	Menu otomatis terhapus dari ringkasan pesanan	Pass
		TC10-04	Menambahkan menu	Klik "Tambah Menu"	Kembali ke halaman menu tanpa kehilangan isi keranjang.	Kembali ke halaman menu tanpa kehilangan isi keranjang.	Pass
		TC10-05	Memilih metode pengantaran "Pesan Antar"	Klik "Pesan Antar"	Menampilkan inputan nama , nomor WhatsApp, catatan, dan diantar ke?.	Menampilkan inputan nama , nomor WhatsApp, catatan, dan diantar ke?.	Pass
		TC10-06	Memilih metode pengantaran "Ambil Sendiri"	Klik "Ambil Sendiri"	Menampilkan inputan nama , nomor WhatsApp, dan catatan.	Menampilkan inputan nama , nomor WhatsApp, dan catatan.	Pass
		TC10-07	Konfirmasi pesanan dengan form kosong		Tombol bayar disable	Tombol bayar disable	Pass

Tabel 5. 48 Pengujian *Payment Gateway*

Kode Test Skenario	Test Skenario	Kode Test Case	Test Case	Test Data	Expested Result	Actual Result	Pass/Fail
TS-11	Cek Fungsi Menyediakan Payment Gateway	TC11-01	Pembayaran berhasil via payment gateway		Pembeli diarahkan ke halaman "Status Pesanan" dengan status "Menunggu Diproses".	Pembeli diarahkan ke halaman "Status Pesanan" dengan status "Menunggu Diproses".	Pass
		TC11-02	Pengguna Menutup <i>Pop-up Payment Gateway</i>		Muncul notifikasi "Menunggu Pembayaran"	Muncul notifikasi "Menunggu Pembayaran"	Pass

Tabel 5. 49 Pengujian Mengubah status Pesanan

Kode Test Skenario	Test Skenario	Kode Test Case	Test Case	Test Data	Expested Result	Actual Result	Pass/Fail
TS-12	Cek Fungsi Status Pesanan	TC12-01	Status berubah dari <i>Menunggu Diproses</i> menjadi <i>Sedang Diproses</i>		Status pada halaman diperbarui menjadi Sedang Diproses dan menampilkan timer estimasi waktu pesanan.	Status pada halaman diperbarui menjadi Sedang Diproses dan menampilkan estimasi waktu pesanan.	Pass
		TC12-02	Status berubah dari <i>Sedang Diproses</i> menjadi <i>Pesanan Selesai</i>		Status pada halaman diperbarui menjadi Pesanan Selesai; timer tidak tampil dan diganti tulisan "Selesai".	Status pada halaman diperbarui menjadi Pesanan Selesai; timer tidak tampil dan diganti tulisan "Selesai".	Pass
		TC12-03	Timer habis saat status Sedang Diproses		Muncul alert Mohon maaf, pesanan lebih lama dari estimasi, Tunggu sebentar lagi!"; status tetap Sedang Diproses; timer berhenti di 00:00	Muncul alert Mohon maaf, pesanan lebih lama dari estimasi, Tunggu sebentar lagi!"; status tetap Sedang Diproses; timer berhenti di 00:00	Pass
		TC12-04	Klik Pesan Lagi pada status Pesanan Selesai		Pesanan Selesai Pengguna diarahkan ke halaman Beranda.	Pesanan Selesai Pengguna diarahkan ke halaman Beranda.	Pass

Tabel 5. 50 Pengujian Mengelola Menu

Kode Test Skenario	Test Skenario	Kode Test Case	Test Case	Test Data	Expested Result	Actual Result	Pass/Fail
TS-13	Cek Fungsi Mengelola Menu	TC13-01	Menampilkan semua menu saat halaman dimuat		Menampilkan semua daftar menu.	Menampilkan semua daftar menu.	Pass
		TC13-02	Menambahkan menu baru	Gambar: "ayam.png" Nama Menu: "Ayam penyet"	Pesan sukses "Berhasil"	Pesan sukses "Berhasil"	Pass

Kode Test Skenario	Test Skenario	Kode Test Case	Test Case	Test Data	Expeded Result	Actual Result	Pass/Fail
				Harga: "Rp. 13.000" Deskripsi: "Nasi, ayam,sambal" Waktu Penyajian (Menit): "9"	menambahkan menu".	menambahkan menu".	
		TC13-03	Memastikan data muncul di form ubah		Unggah Gambar, Nama Menu, Harga, Deskripsi, Waktu Penyajian.	Unggah Gambar, Nama Menu, Harga, Deskripsi, Waktu Penyajian.	Pass
		TC13-04	Mengubah data menu	Gambar: "ayam.png" Nama Menu: "Ayam penyet" Harga: "Rp. 12.000" Deskripsi: "Nasi, ayam,sambal" Waktu Penyajian (Menit): "8"	Pesan sukses "Berhasil mengubah menu".	Pesan sukses "Berhasil mengubah menu".	Pass
		TC13-05	Menghapus menu	(konfirmasi "Ya, Hapus")	Pesan sukses "Berhasil Menghapus Menu".	Pesan sukses "Berhasil Menghapus Menu".	Pass
		TC13-06	Membatalkan penghapusan menu	(konfirmasi "Batal")	Menu tidak terhapus dan tetap ada di dalam daftar.	Menu tidak terhapus dan tetap ada di dalam daftar.	Pass
		TC13-07	Melihat detail menu	Klik ikon mata pada salah satu menu	Menampilkan detail menu (Nama Menu, Harga, Deskripsi, Waktu Penyajian, tombol Ubah & Hapus)	Menampilkan detail menu (Nama Menu, Harga, Deskripsi, Waktu Penyajian, tombol Ubah & Hapus)	Pass
		TC13-08	Buka halaman dashboard	Data: Total Pesanan = 0, Total Menu = 3, Pendapatan = 0	Dashboard menampilkan card dengan angka sesuai data (0, 3, 0)	Dashboard menampilkan card dengan angka sesuai data (0, 3, 0)	Pass

Tabel 5. 51 Pengujian Mengirim Notifikasi Pesanan Masuk

Kode Test Skenario	Test Skenario	Kode Test Case	Test Case	Test Data	Expeded Result	Actual Result	Pass/Fail
TS-14	Cek Fungsi Notifikasi Pesanan Masuk (Badge Sidebar)	TS14-01	Menampilkan badge saat ada pesanan baru	Server memiliki 5 pesanan berstatus "Menunggu Diproses"	Badge pada menu Pesanan menampilkan angka 5	Badge pada menu Pesanan menampilkan angka 5	Pass
		TS14-02	Menyembunyikan badge saat tidak ada pesanan baru	Server mengembalikan jumlah "0"	Badge tidak tampil	Badge tidak tampil	Pass

Tabel 5. 52 Pengujian Menampilkan Daftar dan Detail Pesanan

Kode Test Skenario	Test Skenario	Kode Test Case	Test Case	Test Data	Expeded Result	Actual Result	Pass/Fail
TS-15	Cek Fungsi Mengelola Pesanan	TC15-01	Menampilkan semua data pesanan saat halaman dimuat		Menampilkan semua daftar pesanan.	Menampilkan semua daftar pesanan.	Pass

Kode Test Skenario	Test Skenario	Kode Test Case	Test Case	Test Data	Expested Result	Actual Result	Pass/Fail
		TC15-02	Menampilkan detail pesanan	Klik ikon mata pada salah satu pesanan	Menampilkan halaman detail pesanan yang dipilih	Menampilkan halaman detail pesanan yang dipilih	Pass
		TC15-03	Mengubah status pesanan dari Menunggu Diproses menjadi Pesanan Diproses	Klik tombol Proses Pesanan	Status pesanan berubah menjadi Pesanan Diproses	Status pesanan berubah menjadi Pesanan Diproses	Pass
		TC15-04	Mengubah status pesanan dari Pesanan Diproses menjadi Siap Diambil/ Siap Diantar	Klik tombol Pesanan Siap Diambil/ Siap Diantar dan konfirmasi "Sudah"	Status pesanan berubah menjadi <i>Siap Diambil / Siap Diantar</i>	Status pesanan berubah menjadi <i>Siap Diambil / Siap Diantar</i>	Pass
		TC15-05	Membatalkan perubahan status saat konfirmasi <i>Siap Diambil</i>	Klik tombol Pesanan Siap Diambil/ Siap Diantar dan konfirmasi "Batal"	Status pesanan tetap <i>Pesanan Diproses</i>	Status pesanan tetap <i>Pesanan Diproses</i>	Pass
		TC15-06	Mengubah status pesanan dari <i>Siap Diambil/ Siap Diantar</i> menjadi <i>Pesanan Selesai</i>	Klik tombol Pesanan Sudah Diambil/ Sudah Diantar	Status pesanan berubah menjadi Pesanan Selesai	Status pesanan berubah menjadi Pesanan Selesai	Pass
		TC15-07	Menampilkan semua riwayat pesanan saat dalam dimuat		Menampilkan semua daftar Riwayat pesanan.	Menampilkan semua daftar Riwayat pesanan.	Pass
		TC15-08	Menampilkan detail riwayat pesanan	Klik ikon mata pada salah satu riwayat pesanan	Menampilkan halaman detail riwayat pesanan yang dipilih	Menampilkan halaman detail riwayat pesanan yang dipilih	Pass

Tabel 5. 53 Pengujian Menampilkan Tipe Pengantaran

Kode Test Skenario	Test Skenario	Kode Test Case	Test Case	Test Data	Expested Result	Actual Result	Pass/Fail
TS-16	Cek Fungsi Menampilkan Tipe Pengantaran	TS16-01	Menampilkan label "Ambil sendiri" di Detail Pesanan (penjual)	tipe_pengantaran = "ambil_sendiri"	Halaman Detail Pesanan menampilkan label "Ambil sendiri"	Halaman Detail Pesanan menampilkan label "Ambil sendiri"	Pass
		TS16-02	Menampilkan "Diantar ke" beserta diantar ke di Detail Pesanan (penjual)	tipe_pengantaran = "diantar", diantar_ke = "Depan perpus"	Halaman Detail Pesanan menampilkan Depan perpus	Halaman Detail Pesanan menampilkan Depan perpus	Pass

Tabel 5. 54 Pengujian Halaman Beranda

Kode Test Skenario	Test Skenario	Kode Test Case	Test Case	Test Data	Expested Result	Actual Result	Pass/Fail
TS-17	Cek Fungsi Halaman Beranda	TC17-01	Cari kios atau menu	Ketik "Ayam"	Menampilkan kios yang memiliki menu ayam	Menampilkan kios yang memiliki menu ayam	Pass

Kode Test Skenario	Test Skenario	Kode Test Case	Test Case	Test Data	Expested Result	Actual Result	Pass/Fail
		TC17-02	Menampilkan 5 menu terbaru saat halaman dimuat		Menampilkan 5 menu terbaru	Menampilkan 5 menu terbaru	Pass
		TC17-03	Menampilkan 8 kios terbaru saat halaman dimuat		Menampilkan 8 kios terbaru	Menampilkan 10 kios terbaru	Pass

Berdasarkan pengujian yang telah dilakukan, berikut merupakan rekapitulasi hasil dari pengujian dengan menggunakan *Black Box Testing*.

Tabel 5. 55 Rekapitulasi Pengujian

No	Test Scenario	Jumlah Test Case	Presentase Lulus Uji
1	Cek Fungsi Register	5	100%
2	Cek Fungsi Register Toko	4	100%
3	Cek Fungsi Verifikasi Akun Dengan OTP	3	100%
4	Cek Fungsi Login	3	100%
5	Cek Fungsi Lupa Kata Sandi	6	100%
6	Cek Fungsi Melengkapi dan Mengubah Profil	2	100%
7	Cek Fungsi Logout	1	100%
8	Cek Fungsi Kios	5	100%
10	Cek Fungsi Konfirmasi Pesanan	7	100%
11	Cek Fungsi Menyediakan Payment Gateway	2	100%
12	Cek Fungsi Status Pesanan	4	100%
13	Cek Fungsi Mengelola Menu	8	100%
14	Cek Fungsi Notifikasi Pesanan Masuk (Badge Sidebar)	2	100%
15	Cek Fungsi Mengelola Pesanan	8	100%
16	Cek Fungsi Menampilkan Tipe Pengantaran	2	100%
17	Cek Fungsi Halaman Beranda	3	100%

Dari rekapitulasi pengujian Black Box Testing, diperoleh total 17 skenario uji, di mana semuanya dinyatakan lulus dengan persentase 100%. Hasil ini menunjukkan bahwa fitur-fitur pada aplikasi berfungsi optimal dan sesuai kebutuhan.

5.2.2 Deployment

Tahap *deployment* pada sisi *front-end* bertujuan agar aplikasi PasMeal dapat diakses secara online baik oleh penjual maupun pembeli. Pada proyek ini digunakan *platform* Vercel yang sudah terintegrasi dengan GitHub, sehingga mendukung proses otomatis (*Continuous Deployment*) setiap kali ada perubahan kode.

5.2.2.1 Alur Deployment

Proses *deployment* menggunakan mekanisme *Continuous Deployment* (CD) bawaan Vercel dengan langkah sebagai berikut:

1. Menghubungkan Repositori

Aplikasi dihubungkan dengan repositori GitHub Dashboard_PasMeal dan PasMeal melalui dashboard Vercel. Dengan begitu, setiap perubahan kode di repositori bisa langsung dipantau oleh Vercel.

2. Konfigurasi Build

Pengaturan build dilakukan melalui dashboard Vercel. Untuk proyek React, konfigurasi yang digunakan adalah:

- *Build command*: npm run build (Perintah ini akan membuat versi produksi aplikasi).
- *Output directory*: dist/ (Hasil build disimpan di folder ini dan nantinya dipublikasikan ke server).

3. Proses Deployment Otomatis

Setiap kali ada kode baru yang di-*push* ke branch utama (main), Vercel akan secara otomatis:

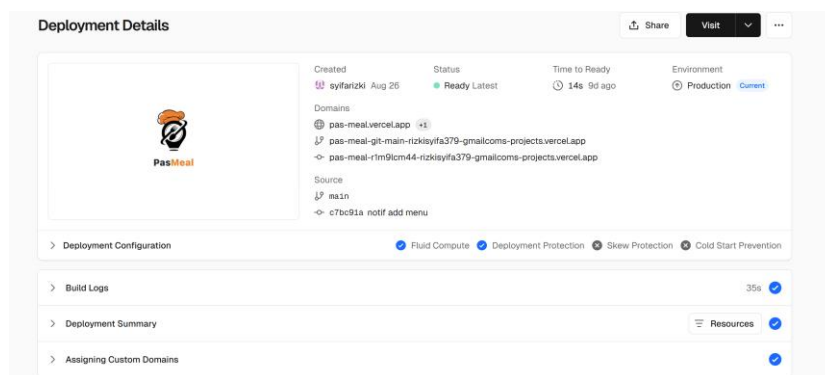
- Mengambil kode terbaru dari GitHub.
- Menjalankan npm install untuk memasang dependensi dan npm run build untuk membuat aplikasi siap produksi.
- Mengunggah hasil build di folder dist/ ke jaringan server global (CDN) Vercel. Dengan cara ini, aplikasi bisa langsung diakses melalui domain default dari Vercel

5.2.2.2 Hasil Deployment

Sesudah proses deployment di Vercel selesai, aplikasi *front-end* PasMeal sudah berhasil dipublikasikan dan dapat diakses secara umum melalui URL produksi yang tersedia.

Tabel 5. 56 Hasil Deployment

Aplikasi	Environment	URL
Dashboard Penjual	Production	https://pas-meal.vercel.app/
Aplikasi Pembeli	Production	https://pas-meal-2rlb.vercel.app/



Gambar 5. 69 Hasil Deployment Dashboard Penjual

Deployment Details

[Share](#)[Visit](#)[...](#)



PasMeal

Created	Status	Time to Ready	Environment
 syfarizki Aug 26	Ready Latest	 14s 9d ago	 Production Current

Domains

-  pas-meal.vercel.app [+1](#)
-  pas-meal-git-main-rizkisyfa379-gmailcoms-projects.vercel.app
-  pas-meal-r1m9lcm44-rizkisyfa379-gmailcoms-projects.vercel.app

Source

-  main
-  c7bc91a notif add menu

> Deployment Configuration

 Fluid Compute  Deployment Protection  Skew Protection  Cold Start Prevention

> Build Logs

35s 

> Deployment Summary

[Resources](#) 

> Assigning Custom Domains



Gambar 5. 70 Hasil Deployment Aplikasi Pembeli

BAB 6

KESIMPULAN DAN SARAN

Bab ini menjelaskan mengenai kesimpulan yang dapat diambil dari hasil penelitian yang dilakukan serta saran yang diberikan untuk pengembangan selanjutnya.

6.1 Kesimpulan

Berdasarkan hasil perancangan, implementasi, dan pengujian *Front-End* Web Pemesanan Makanan *Online* menggunakan React JS (Studi Kasus: Kantin Universitas Pasundan Setiabudi), maka dapat disimpulkan bahwa:

1. Melalui tugas akhir ini, telah berhasil dikembangkan sebuah tampilan antarmuka (*front-end*) untuk aplikasi pemesanan makanan di Kantin Universitas Pasundan Setiabudi menggunakan React JS. Aplikasi ini menyediakan antarmuka fungsional bagi mahasiswa (pembeli) dan *dashboard* pengelolaan bagi pemilik kios (penjual).
2. Penggunaan arsitektur berbasis komponen pada React JS menciptakan pengalaman pengguna yang interaktif, di mana fitur seperti keranjang belanja dan pengelolaan menu dapat diperbarui secara *real-time* tanpa perlu memuat ulang halaman.
3. Metodologi pengembangan Scrum memungkinkan proses pengerjaan yang iteratif dan terkelola, sehingga produk *front-end* dapat dikembangkan secara bertahap mulai dari perancangan antarmuka hingga terintegrasi dengan API di sisi *back-end*.

6.2 Saran

Berdasarkan hasil penelitian dan pengembangan yang telah dilakukan, berikut adalah beberapa saran yang dapat dijadikan pertimbangan untuk pengembangan aplikasi di masa mendatang:

1. Pada fitur Riwayat Pesanan, disarankan menambahkan fasilitas pencarian berdasarkan nama pembeli serta filter berdasarkan periode pesanan (harian, mingguan, atau bulanan). Hal ini akan memudahkan penjual dalam menemukan pesanan tertentu dan menyusun laporan penjualan secara lebih teratur.
2. Dalam pengembangan aplikasi, disarankan melakukan uji coba antarmuka dengan melibatkan pengguna (pembeli dan penjual) agar tampilan dan alur aplikasi sesuai kebutuhan serta mudah dipahami, sekaligus memperoleh masukan untuk penyempurnaan.
3. Disarankan agar aplikasi dilengkapi dengan indikator pemuatan (*loading*) saat data ditampilkan. Hal ini bertujuan memberikan kejelasan kepada pengguna bahwa sistem sedang memproses data, sehingga menghindari anggapan adanya kesalahan pada aplikasi.

DAFTAR PUSTAKA

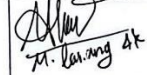
- [ANG23] Anggara, D., Ambaradewi, N.L.G., & Bawantara, I.P.Y., “Aplikasi Pemesanan Menu Pada Edery Cafe Berbasis Android”, Jurnal Manajemen Dan Teknologi Informasi, Vol. 13, No. 1, 2023.
- [CHA20] Chastro, C., & Darmawan, E., “Perbandingan Pengembangan Front End Menggunakan Blade Template dan Vue Js”, Jurnal Strategi, Volume 2, Nomor 2, November 2020.
- [FER22] Ferdiansyah, Muhammad, “Pembuatan Frontend Pada Proyek Pengembangan Website Sekolah Di Inagata Technosmith”, tersedia: Februari 2025, <https://repository.dinamika.ac.id/id/eprint/6569/>, Agustus 2022.
- [GIN22] Ginting, I.D.Y.T.A., Tolle, H., & Brata, K.C., “Pengembangan Sistem Manajemen Sampah TPS3R Dinas Lingkungan Hidup Kota Batu berbasis Web Responsive menggunakan Tailwind dan ReactJS”, Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer, Vol. 6, No. 12, Desember 2022.
- [GUS20] Gustama, Reinaldi, “Implementasi Scrum Pada Manajemen Proyek Pengembangan Aplikasi Sistem Monitoring Dan Evaluasi Pembangunan (Smep)”, tersedia: Mei 2025, <https://dspace.uui.ac.id/handle/123456789/31414>, 2020.
- [HAD24] Hadi, A.P., “Mengenal Frontend Development”, Yayasan Prima Agus Teknik, September 2024.
- [JUA23] Juan, K., & Budi, S., “Pengembangan Menu Digital Menggunakan ReactJs Implementasi Hasil Belajar Studi Independen di Frontend Engineering Program Ruangguru CAMP (Career Acceleration Bootcamp)”, Jurnal Strategi, Volume 5, Nomor 1, Mei 2023.
- [KAN23] Kania, D.S., Afirahmi, D.M., Nuria, D.P., & Yusup, D., “Perancangan dan Pengembangan Aplikasi E(AT)VERYDAY sebagai Sistem Pemesanan Kantin Berbasis Website”, Journal Of Social Science Research, Volume 3, Nomor 3, 2023.
- [KUK22] Kukuh, Puji, Lestari, “Implementasi Perancangan Desain Frontend Pada Aplikasi Monitoring Pemasangan Jaringan Menggunakan Arsitektur Microservice Pada Pt Lintas Data Multimedia”, tersedia: Mei 2025, <https://repository.polinela.ac.id/2704/>, Oktober 2022.
- [MAU23] Maurya, P., Keshari, P., Kumar, P., & Katiyar, A., “Web Development Using ReactJS”, Journal of Current Research in Engineering and Science, Volume 6, August 2023.
- [MUS24] Musyaffa, A.I., Zulfa, M.I., Alim, M.S., “Rancang Bangun Purecompute Platform E-Commerce untuk Belanja Laptop Berbasis Website”, Jurnal Sistem Informasi dan Teknologi Komputasi, Volume 1, Nomor 1, Januari 2024.
- [MUS24] Musyaffa, M.N., Fajri, H., & Ikhsan, S.H.A., “Pembuatan Aplikasi Pemesanan Dan Pemasangan Kaca Film Mobil Berbasis Android Dan Website (Studi Kasus Cv. Agni Sentral Motor)”, JATI (Jurnal Mahasiswa Teknik Informatika), Vol. 8, No. 4, Agustus 2024

- [NEY24] Neydeline, Dorna, “Pembangunan Website Pengelolaan Kegiatan Gereja Kemah Injil Indonesia (Gkii) Tanjung Lapang”, tersedia: Mei 2025, <https://e-journal.uajy.ac.id/32747/>, Oktober 2024.
- [NUR19] Nurlaila, F., “Aplikasi Pemesanan Makanan pada Restoran 1953 Indonesia Berbasis Web”, Jurnal Informatika Universitas Pamulang, Vol. 4, No. 1, Maret 2019.
- [RIZ24] Rizki, L., & Tresnawati, Y., “Rancang Bangun Aplikasi Pelayanan Pemesanan Makanan Dan Minuman Berbasis Website”, Indonesian Journal Computer Science, Vol. 3, No. 1, April 2024.
- [SAC16] Sachdeva, S., “Scrum Methodology”, International Journal Of Engineering And Computer Science, Volume 5, 6 June 2016.
- [SAL23] Salmi, H.A., “Comparative CSS frameworks”, Multi-Knowledge Electronic Comprehensive Journal For Education And Science Publications (MECSJ), 2023.
- [SAT19] Satria., & Seabtian, D.J., “Sistem Informasi E-Marketplace Pada Pemesanan Jasa Fotografi Berbasis Web Di Kotawaringin Timur”, Jurnal Penelitian Dosen Fikom (UNDA), Vol. 10, No. 2, November 2019.
- [SAZ24] Sazli, W., Wahyuni, L., “Implementasi E-Canteen menggunakan Metode FIFO berbasis Android pada PT. Mahzar Jaya”, INNOVATIVE: Journal Of Social Science Research, Volume 4, Nomor 6, 2024.
- [SCH20] Schwaber, K., & Sutherland, J., “The Scrum Guide”, 2020.
- [TJA24] Tjandra, J.C., Sidhi, T.A.P., & Sundoro, B.T., “Pembangunan Sistem Informasi Manajemen Hotel Semeru Berbasis Websitedan Mobile”, Jurnal Informatika Atma Jogja, Vol. 5, No. 1, 2024.
- [WIR20] Wirfs-Brock, Allen, & Eich, Brendan, “JavaScript: The First 20 Years”, tersedia : Februari 2025, <https://dl.acm.org/doi/abs/10.1145/3386327> , Juni 2020.
- [ZAH24] Zahrani, Sabina, izatul, “Sistem Informasi Toko Roti Berbasis Website”, tersedia : Februari 2025, <http://eprints.poltektegal.ac.id/4379/>, September 2024.

LAMPIRAN A

BERITA ACARA WAWANCARA

1. Wawancara Pembeli

 Berita Acara Wawancara Tugas Akhir Program Studi Teknik Informatika, Fakultas Teknik, Universitas Pasundan			
Studi Kasus Pemesanan di Kantin Unpas Setiabudi Wawancara Pelanggan			
Hari, tanggal : Selasa, 25/01/2025.			
Nama : M. Nurang Abid K.			
Pertanyaan			
1. Apakah Anda pernah mengalami kesalahan dalam pemesanan (misalnya pesanan salah atau terlambat)?			
Jawab : Pernah			
2. Seberapa sering Anda merasa harus menunggu lama untuk mendapatkan pesanan?			
Jawab : Cukup Sering.			
3. Apakah Anda ingin mendapatkan estimasi waktu yang lebih akurat saat menunggu makanan siap?			
Jawab : Tentu, Ingin.			
4. Menurut Anda, apakah website pemesanan bisa membantu membuat proses pemesanan lebih cepat dan mudah?			
Jawab : Ya, apps bisa membantu			
5. Apakah Anda lebih suka melakukan pemesanan online atau langsung datang ke kantin?			
Jawab : Secara Pribadi, jika ada platform nya saya lebih suka online.			
6. Fitur apa yang paling Anda inginkan dalam sistem pemesanan online?			
Jawab : Mungkin fitur yang ada di gofood.			
7. Jika pemesanan dilakukan melalui website, apakah Anda merasa nyaman dengan pembayaran digital?			
Jawab : tentu, nyaman & saya.			
8. Bagaimana menurut Anda jika ada sistem yang memungkinkan Anda memesan makanan dulu, lalu mengambilnya saat sudah siap tanpa perlu menunggu lama?			
Jawab : Jika hal itu diterapkan di kantin kampus, itu sangat bagus.			
Peneliti		Narasumber	
 Syifa	 Risma	 Putri	 M. Nurang Abid K.

Gambar A. 1 Berita Acara Wawancara Dengan Pembeli (1)

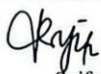
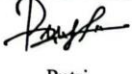
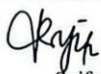
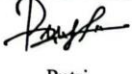
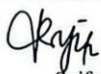
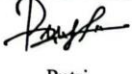
 Berita Acara Wawancara Tugas Akhir Program Studi Teknik Informatika, Fakultas Teknik, Universitas Pasundan Studi Kasus Pemesanan di Kantin Unpas Setiabudi Wawancara Pelanggan									
Hari, tanggal : 25 Februari 2025, Selasa Nama : M. Rivan Sahronie									
Pertanyaan									
1. Apakah Anda pernah mengalami kesalahan dalam pemesanan (misalnya pesanan salah atau terlambat)?	Jawab : Pernah								
2. Seberapa sering Anda merasa harus menunggu lama untuk mendapatkan pesanan?	Jawab : tidak terlalu sering								
3. Apakah Anda ingin mendapatkan estimasi waktu yang lebih akurat saat menunggu makanan siap?	Jawab : Bekerja								
4. Menurut Anda, apakah website pemesanan bisa membantu membuat proses pemesanan lebih cepat dan mudah?	Jawab: Bisa . tergantung dari sistem yang dibuat								
5. Apakah Anda lebih suka melakukan pemesanan online atau langsung datang ke kantin?	Jawab : online								
6. Fitur apa yang paling Anda inginkan dalam sistem pemesanan online?	Jawab : pembayaran secara digital								
7. Jika pemesanan dilakukan melalui website, apakah Anda merasa nyaman dengan pembayaran digital?	Jawab : Iya .								
8. Bagaimana menurut Anda jika ada sistem yang memungkinkan Anda memesan makanan dulu, lalu mengambilnya saat sudah siap tanpa perlu menunggu lama?	Jawab : Fitur yg bagus								
<table border="1"> <thead> <tr> <th colspan="3">Peneliti</th> <th>Narasumber</th> </tr> </thead> <tbody> <tr> <td> Syifa</td> <td> Risma</td> <td> Putri</td> <td></td> </tr> </tbody> </table>		Peneliti			Narasumber	 Syifa	 Risma	 Putri	
Peneliti			Narasumber						
 Syifa	 Risma	 Putri							

Gambar A. 2 Berita Acara Wawancara Dengan Pembeli (2)

 Berita Acara Wawancara Tugas Akhir Program Studi Teknik Informatika, Fakultas Teknik, Universitas Pasundan			
Studi Kasus Pemesanan di Kantin Unpas Setiabudi Wawancara Pelanggan			
Hari, tanggal : Selasa, 25-02-2025 Nama : Widya Dwi Indrianti			
Pertanyaan			
1. Apakah Anda pernah mengalami kesalahan dalam pemesanan (misalnya pesan salah atau terlambat)?			
Jawab : Pernah			
2. Seberapa sering Anda merasa harus menunggu lama untuk mendapatkan pesanan?			
Jawab : Cukup sering			
3. Apakah Anda ingin mendapatkan estimasi waktu yang lebih akurat saat menunggu makanan siap?			
Jawab : Iya			
4. Menurut Anda, apakah website pemesanan bisa membantu membuat proses pemesanan lebih cepat dan mudah?			
Jawab: Iya bisa			
5. Apakah Anda lebih suka melakukan pemesanan online atau langsung datang ke kantin?			
Jawab : Pemesanan online, karna tidak memakan waktu			
6. Fitur apa yang paling Anda inginkan dalam sistem pemesanan online?			
Jawab : pemiihan menu makanan / minuman			
7. Jika pemesanan dilakukan melalui website, apakah Anda merasa nyaman dengan pembayaran digital?			
Jawab : Iya, karena pembayaran digital lebih simple			
8. Bagaimana menurut Anda jika ada sistem yang memungkinkan Anda memesan makanan dulu, lalu mengambilnya saat sudah siap tanpa perlu menunggu lama?			
Jawab : menurut saya akan sangat berguna			
Peneliti		Narasumber	
 Syifa	 Risma	 Putri	 Narasumber

Gambar A. 3 Berita Acara Wawancara Dengan Pembeli (3)

2. Wawancara Penjual

 Berita Acara Wawancara Tugas Akhir Program Studi Teknik Informatika, Fakultas Teknik, Universitas Pasundan Studi Kasus Pemesanan di Kantin Unpas Setiabudi Wawancara Penjual											
Hari, tanggal : Rabu, 26.02, 2025 Nama : Aldo Setiawan (Dapur Uje - 39)											
Pertanyaan											
1. Bagaimana cara pelanggan memesan makanan di tempat Anda saat ini? (secara langsung/chat)											
Jawab : banyak chat											
2. Apakah pernah terjadi kesalahan dalam mencatat pesanan, seperti pesanan tertukar atau tidak tercatat? Jika iya, apakah pelanggan ada yang protes?											
Jawab : Salah bikin sama tertukar											
3. Apakah ada kesulitan saat mengantarkan pesanan, seperti menemukan nomor meja tempat pelanggan duduk?											
Jawab : Pelanggan nunggu											
4. Ketika pelanggan sedang ramai, bagaimana Anda mengatur antrian pemesanan? Misalnya, jika pelanggan pertama memesan dalam jumlah banyak dan pelanggan berikutnya hanya memesan sedikit, pesanan mana yang akan dibuat lebih dulu?											
Jawab : Yang pertama / digabungin											
5. Biasanya, butuh waktu berapa lama untuk menyiapkan satu pesanan?											
Jawab : 15 - 20 menit											
6. Apakah pelanggan sering bertanya kapan pesanan mereka akan siap?											
Jawab : gada											
7. Saat ini, bagaimana Anda mencatat dan menyimpan daftar pesanan pelanggan ketika sedang ramai? Apakah dicatat di kertas, atau hanya diingat saja?											
Jawab : di catet											
8. Apakah Anda tertarik jika dibuatkan website atau aplikasi untuk mempermudah mengelola pemesanan?											
Jawab :											
<table border="1"> <thead> <tr> <th colspan="3">Peneliti</th> <th>Narasumber</th> </tr> </thead> <tbody> <tr> <td> Syifa</td> <td> Risma</td> <td> Putri</td> <td></td> </tr> </tbody> </table>				Peneliti			Narasumber	 Syifa	 Risma	 Putri	
Peneliti			Narasumber								
 Syifa	 Risma	 Putri									

Gambar A. 4 Berita Acara Wawancara Dengan Penjual (1)

 Berita Acara Wawancara Tugas Akhir Program Studi Teknik Informatika, Fakultas Teknik, Universitas Pasundan Studi Kasus Pemesanan di Kantin Unpas Setiabudi Wawancara Penjual			
Hari, tanggal : Rabu, 26, 02, 2021 Nama : Ria (Wawancara nomor 066y - 49)			
Pertanyaan			
1. Bagaimana cara pelanggan memesan makanan di tempat Anda saat ini? (secara langsung/chat)			
Jawab : langsung			
2. Apakah pernah terjadi kesalahan dalam mencatat pesanan, seperti pesanan tertukar atau tidak tercatat? Jika iya, apakah pelanggan ada yang protes?			
Jawab : Pernah			
3. Apakah ada kesulitan saat mengantarkan pesanan, seperti menemukan nomor meja tempat pelanggan duduk?			
Jawab : jarang			
4. Ketika pelanggan sedang ramai, bagaimana Anda mengatur antrian pemesanan? Misalnya, jika pelanggan pertama memesan dalam jumlah banyak dan pelanggan berikutnya hanya memesan sedikit, pesanan mana yang akan dibuat lebih dulu?			
Jawab : berurutan			
5. Biasanya, butuh waktu berapa lama untuk menyiapkan satu pesanan?			
Jawab : 1 menit			
6. Apakah pelanggan sering bertanya kapan pesanan mereka akan siap?			
Jawab : jarang			
7. Saat ini, bagaimana Anda mencatat dan menyimpan daftar pesanan pelanggan ketika sedang ramai? Apakah dicatat di kertas, atau hanya diingat saja?			
Jawab : di hafal			
8. Apakah Anda tertarik jika dibuatkan website atau aplikasi untuk mempermudah mengelola pemesanan?			
Jawab : Iya boleh			
Peneliti		Narasumber	
 Syifa	 Risma	 Putri	 Narasumber

Gambar A. 5 Berita Acara Wawancara Dengan Penjual (2)

 Berita Acara Wawancara Tugas Akhir Program Studi Teknik Informatika, Fakultas Teknik, Universitas Pasundan											
Studi Kasus Pemesanan di Kantin Unpas Setiabudi Wawancara Penjual											
Hari, tanggal : <u>Rabu, 26, 02, 2025</u> Nama : <u>Diva (SO - Salel Drinks)</u>											
Pertanyaan											
1. Bagaimana cara pelanggan memesan makanan di tempat Anda saat ini? (secara langsung/chat)											
Jawab : <u>gaada chat</u>											
2. Apakah pernah terjadi kesalahan dalam mencatat pesanan, seperti pesanan tertukar atau tidak tercatat? Jika iya, apakah pelanggan ada yang protes?											
Jawab : <u>Pernah salah bikin</u>											
3. Apakah ada kesulitan saat mengantarkan pesanan, seperti menemukan nomor meja tempat pelanggan duduk?											
Jawab : <u>Sering bingung</u>											
4. Ketika pelanggan sedang ramai, bagaimana Anda mengatur antrean pemesanan? Misalnya, jika pelanggan pertama memesan dalam jumlah banyak dan pelanggan berikutnya hanya memesan sedikit, pesanan mana yang akan dibuat lebih dulu?											
Jawab : <u>Yang pertama</u>											
5. Biasanya, butuh waktu berapa lama untuk menyiapkan satu pesanan?											
Jawab : <u>2 menit</u>											
6. Apakah pelanggan sering bertanya kapan pesanan mereka akan siap?											
Jawab : <u>mgga karena cepet</u>											
7. Saat ini, bagaimana Anda mencatat dan menyimpan daftar pesanan pelanggan ketika sedang ramai? Apakah dicatat di kertas, atau hanya diingat saja?											
Jawab : <u>di inget</u>											
8. Apakah Anda tertarik jika dibuatkan website atau aplikasi untuk mempermudah mengelola pemesanan?											
Jawab : <u>Iya</u>											
<table border="1" style="width: 100%; text-align: center;"> <thead> <tr> <th colspan="3">Peneliti</th> <th>Narasumber</th> </tr> </thead> <tbody> <tr> <td> Syifa</td> <td> Risma</td> <td> Putri</td> <td></td> </tr> </tbody> </table>				Peneliti			Narasumber	 Syifa	 Risma	 Putri	
Peneliti			Narasumber								
 Syifa	 Risma	 Putri									

Gambar A. 6 Berita Acara Wawancara Dengan Penjual (3)



Berita Acara Wawancara Tugas Akhir

Program Studi Teknik Informatika, Fakultas Teknik, Universitas Pasundan

Studi Kasus Pemesanan di Kantin Unpas Setiabudi

Wawancara Penjual

Hari, tanggal : Kamis, 20 Februari 2025

Nama : Yayan Dwi Yana

Pertanyaan

1. Bagaimana kondisi kantin UNPAS sebelum dilakukan perubahan dan siapa yang mengelolanya?

Jawab : Sebelumnya, kantin masih kurang terawat dan dikelola oleh Koperasi Warga UNPAS.

2. Setelah dikelola universitas, apa langkah utama yang dilakukan untuk memperbaiki kantin?

Jawab : Universitas melakukan renovasi bangunan agar lebih nyaman dan layak digunakan.

3. Saat ini berapa jumlah kios yang tersedia dan siapa saja yang diprioritaskan dalam penyewaan?

Jawab : Ada sekitar 50 kios. Prioritas pertama untuk pedagang lama, kemudian pensiunan staf/dosen, serta masyarakat sekitar kampus.

4. Bagaimana kondisi kantin sekarang setelah dikelola universitas?

Jawab : Kantin lebih nyaman, tertata, dan mendukung kebutuhan mahasiswa, dosen, staf, serta warga sekitar.

5. Apakah saat ini sudah ada sistem digital yang digunakan dalam pengelolaan kantin?

Jawab : Belum, semua masih dilakukan secara manual.

6. Bagaimana pendapat Bapak jika diterapkan sistem pemesanan makanan online di kantin UNPAS??

Jawab : Sangat mendukung, bisa memudahkan mahasiswa dalam memesan makanan dan penjual dalam mengelola pesanan.

Peneliti			Narasumber
			
Syifa	Risma	Putri	Yayan Dwi Yana

Gambar A. 7 Berita Acara Wawancara Dengan Badan Pengelola Kantin

LAMPIRAN B

DOKUMENTASI



Gambar B. 1 Dokumentasi Dengan Badan Pengelola Kantin



Gambar B. 2 Dokumentasi Dengan Penjual

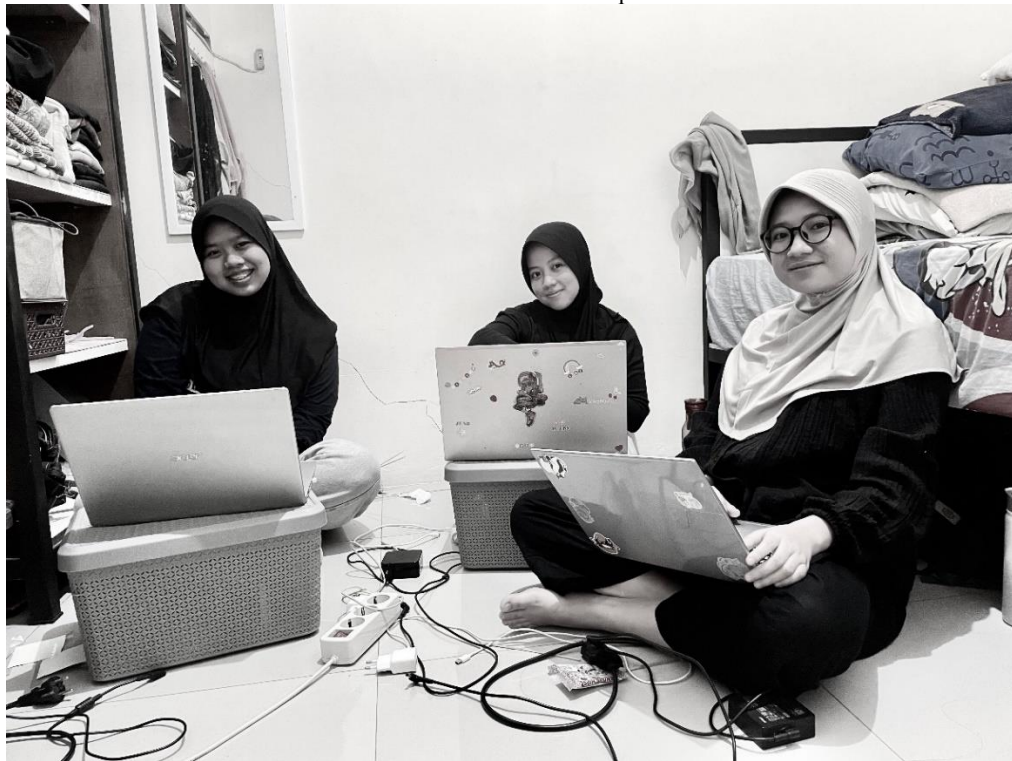


Gambar B. 3 Dokumentasi Dengan Pembeli

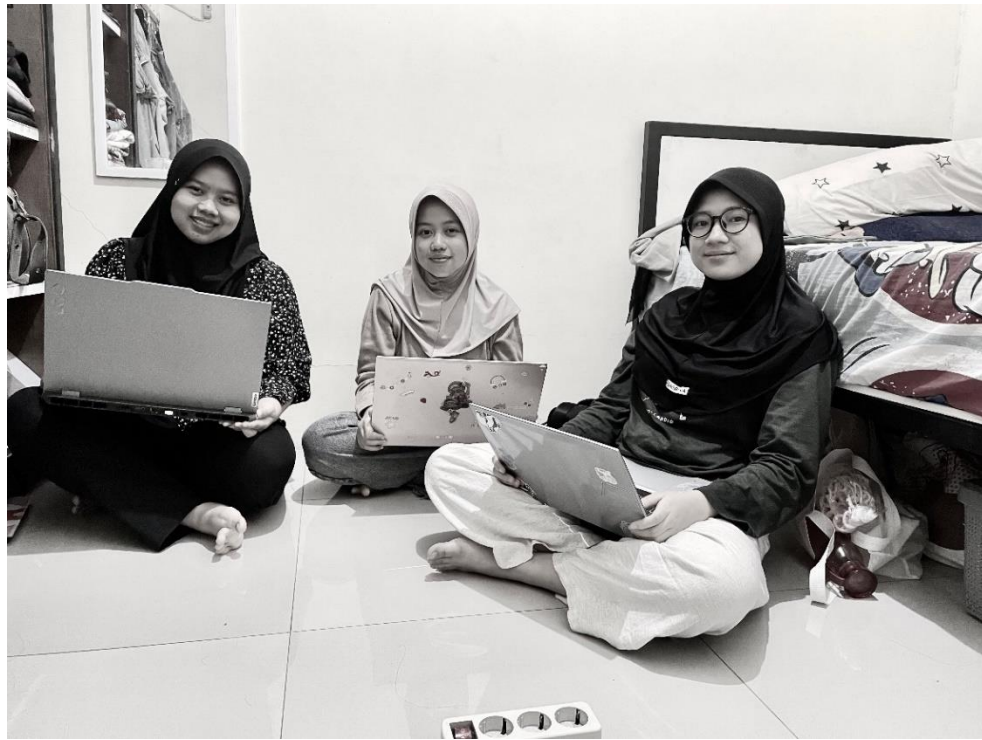
LAMPIRAN C
DOKUMENTASI SCRUM



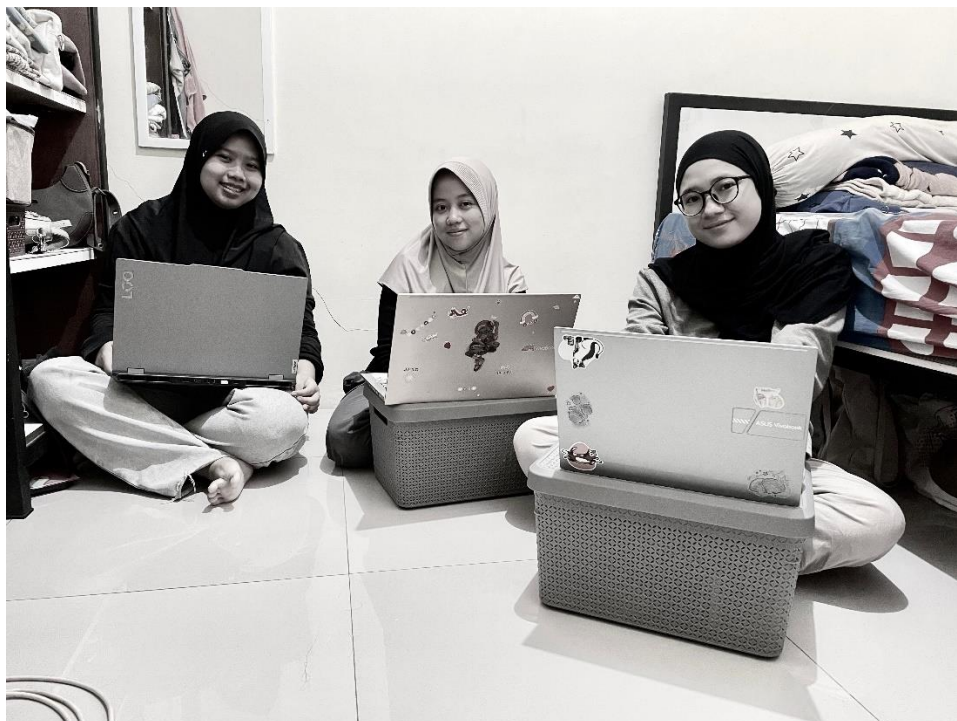
Gambar C. 1 Dokumentasi Sprint 1



Gambar C. 2 Dokumentasi Sprint 2



Gambar C. 3 Dokumentasi Sprint 3



Gambar C. 4 Dokumentasi Sprint 4

LAMPIRAN D

BERITA ACARA DELIVERY

1. Penjual



PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PASUNDAN BANDUNG
Jln.Dr. Setiabudhi no. 193, Bandung 40153

BERITA ACARA

Hari/Tanggal : Rabu, 10 September 2025
Narasumber : Indriani
Agenda : Penyerahan hasil akhir aplikasi yang telah dideploy

Pada hari ini, telah dilaksanakan penyerahan aplikasi PasMeal yang telah berhasil di-deploy. Proses penyerahan dilakukan oleh Syifa Rizki Mutia selaku Pengembang Aplikasi kepada Indriani selaku pihak penjual.

Adapun tautan aplikasi yang diserahkan adalah: <https://pas-meal.vercel.app/>

Berdasarkan hasil uji coba, aplikasi pemesanan makanan online di kantin Universitas Pasundan Setiabudi dinyatakan siap digunakan untuk mendukung kegiatan pemesanan dan transaksi di kantin.

Yang Menghadiri:

No	Nama	Posisi
1	Syifa Rizki Mutia	Front-End Developer
2	Indriani	Penjual

Mengetahui,

Yang Menyerahkan


Syifa Rizki Mutia

Yang Menerima


Indriani

Gambar D. 1 Berita Acara Delivery Penjual

2. Pembeli



PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PASUNDAN BANDUNG
 Jln.Dr. Setiabudhi no. 193, Bandung 40153

BERITA ACARA

Hari/Tanggal : Rabu, 10 September 2025
Narasumber : Widya Dwi Indrianti
Agenda : Penyerahan hasil akhir aplikasi yang telah dideploy

Pada hari ini, telah dilaksanakan penyerahan aplikasi PasMeal yang telah berhasil di-deploy. Proses penyerahan dilakukan oleh Syifa Rizki Mutia selaku Pengembang Aplikasi kepada Widya Dwi Indrianti selaku pihak pembeli.

Adapun tautan aplikasi yang diserahkan adalah: <https://pas-meal-2rlb.vercel.app/>

Berdasarkan hasil uji coba, aplikasi pemesanan makanan online di kantin Universitas Pasundan Setiabudi dinyatakan siap digunakan untuk mendukung kegiatan pemesanan dan transaksi di kantin.

Yang Menghadiri:

No	Nama	Posisi
1	Syifa Rizki Mutia	Front-End Developer
2	Widya Dwi Indrianti	Pembeli

Yang Menyerahkan



Syifa Rizki Mutia

Yang Menerima



Widya Dwi Indrianti

Gambar D. 2 Berita Acara Delivery Pembeli



Gambar D. 3 Dokumentasi Delivery Penjual



Gambar D. 4 Dokumentasi Delivery Pembeli