

BAB 3 PELAKSANAAN KERJA MAGANG

3.1 Kedudukan dan Organisasi

Dalam praktik kerja magang di PT. Solusi Media Ravel Teknologi, penulis diposisikan sebagai *Backend Developer* di bawah supervisi oleh Nicky Fetrayang berkedudukan sebagai *Programmer Web Apps* PT. Solusi Media Ravel Teknologi. Dalam proses kerjanya, Nicky Fetra berperan dalam mengintegrasikan *website* yang telah dibuat. Selain berkoordinasi dengan Nicky Fetra penulis juga bekerja sama dengan Laksa Ersya yang berperan sebagai *UI/UX Designer*, Nicholas Irvan sajoni yang berperan sebagai *Frontend Developer*, dan Fahren Kurniawan yang berperan sebagai *Tester*.

3.2 Tugas yang Dilakukan

Selama praktik magang, penulis memiliki tugas utama untuk membuat *website marketplace* untuk UMKM yaitu karyanoesantara.id dengan fitur yang lengkap untuk menggantikan *website marketplace* yang sebelumnya masih belum berjalan dengan baik. Dalam pengembangan-nya akan digunakan PHP Native dan MariaDB MySQL sebagai sistem *database*-nya. Apabila diperlukan, dapat digunakan *framework* atau teknologi lain untuk memenuhi kebutuhan dalam pengembangan-nya.

3.3 Uraian Pelaksanaan Magang

Pelaksanaan kerja magang di PT. Solusi Media Ravel Teknologi dimulai pada tanggal 19 Mei 2021 - 19 November 2021. Pada awal praktik kerja magang, dimulai dengan kegiatan memperkenalkan profil perusahaan. Kemudian penulis dimasukkan ke dalam tim *web development* pada divisi KARTARA.

Pada minggu pertama, penulis ditugaskan sebagai *Frontend Developer* untuk *website* mitra.karyanoesantara.id. Pada minggu kedua, tugas penulis digantikan menjadi *Backend Developer* untuk *swebsite* karyanoesantara.id. Pemandahan tugas ini dilakukan setelah penulis diusulkan oleh pembimbing lapangan. Tugas utama yang diberikan adalah untuk membuat *website* baru

karyanoesantara.id yang sebelumnya hanya berupa *prototype* dan belum memiliki tampilan yang menarik dan fitur yang lengkap, menjadi *website* yang memiliki tampilan yang baru, lebih menarik, dan fitur yang lengkap.

Dalam perancangannya, *website* memiliki beberapa fitur utama, yaitu masuk dan membuat akun, melihat produk, memesan produk, membayar pesanan, melihat pesanan, dan *chat*. Dalam implementasinya, *website* karyanoesantara.id ini dibangun menggunakan PHP Native dengan menggunakan sistem *database* MariaDB MySQL. Khusus untuk implementasi fitur *chat* dibuat menggunakan Express.js sebagai *framework* dan Socket.IO sebagai *library websocket*, penggunaan *framework* ini disebabkan oleh *framework* PHP Native yang tidak dapat mengimplementasikan fungsi *websocket* yang merupakan fungsi penting dalam aplikasi *chat*.

Pelaksanaan kerja magang dapat diuraikan secara mingguan seperti yang ditampilkan pada Tabel 3.1 berikut.

Minggu Ke -	Kegiatan
1	Memahami cara kerja <i>website</i> mitra.karyanoesantara.id dan melakukan perbaikan <i>responsive</i> pada halaman <i>Dashboard</i> .
2	Pemindahan tugas dari <i>Frontend Developer</i> untuk mitra.karyanoesantara.id menjadi <i>Backend Developer</i> untuk karyanoesantara.id. Melakukan perbaikan <i>responsive</i> pada halaman <i>Dashboard</i> mitra.karyanoesantara.id. Menambahkan tampilan halaman <i>Login</i> dan <i>Register</i> karyanoesantara.id
3	Menampilkan produk pada halaman <i>Home</i> , menambahkan halaman Kategori dan menampilkan sebagian data. Menampilkan informasi pengguna pada halaman Profil. Mengaktifkan fitur <i>login</i> dan <i>register</i> pada halaman <i>Login</i> dan <i>Register</i> . Mengaktifkan fungsi pencarian produk.
4	Menambahkan halaman Bantuan. Fungsi memasukkan produk ke dalam keranjang yang masih belum sempurna. Fitur mengganti informasi pengguna dan mengganti <i>password</i> pengguna. Tambahkan <i>pagination</i> , <i>filter</i> harga, <i>filter</i> kategori, dan <i>sort</i> pada halaman Kategori. Menambahkan halaman Kurir Bangsa.

5	Menambahkan produk dari mitra dan tambahan informasi pada halaman Detail Produk. Fitur <i>upload</i> foto profil pengguna pada halaman Profil. Fitur menambahkan, menampilkan, mengubah, dan menghapus informasi alamat pengguna pada halaman Profil bagian Alamat. Tahap pengembangan proses pemesanan pada halaman <i>Checkout</i> yaitu pemilihan alamat, dan daftar produk yang dipesan.
6	Tahap lanjutan pengembangan halaman <i>Checkout</i> , menampilkan dan mengubah jasa pengiriman, penggunaan voucher, pemilihan metode pembayaran. Mengaktifkan fungsi metode pembayaran Mandiri <i>Virtual Account</i> , Permata <i>Virtual Account</i> , BRI <i>Virtual Account</i> , BNI <i>Virtual Account</i> , QRIS. Menampilkan pesanan pengguna dengan sederhana pada halaman Profil bagian Pesanan Saya.
7	Tahap lanjutan pengembangan halaman Profil bagian Pesanan Saya, menampilkan daftar pesanan pengguna berdasarkan status pesanan setiap pesanan dengan fungsi <i>sort</i> dan <i>search</i> . Menambahkan halaman Hubungi Kami/Bantuan. <i>Bug fix</i> tambah produk ke dalam keranjang, gambar <i>default</i> produk.
8	Menambahkan fitur <i>wishlist</i> produk dan menampilkan daftar <i>wishlist</i> pada halaman Profil bagian Wishlist dengan fungsi pencarian. Memperbaiki tampilan <i>responsive</i> pada halaman <i>Home</i> . <i>Bug fix</i> fungsi <i>register</i> , fungsi menambahkan produk ke dalam keranjang, fungsi <i>edit profile</i> , dan lain-lain.
9	Menambahkan fitur <i>login</i> dan <i>register</i> menggunakan Google dan Facebook dengan bantuan Firebase, dan penyesuaian <i>database</i> . Memperbaiki tampilan <i>responsive</i> pada halaman <i>Home</i> , Pembayaran, Detail Produk, Kategori Produk, <i>Login</i> , Daftar, dan komponen navigasi. Implementasi fungsi verifikasi nomor telepon menggunakan kode OTP dengan memanfaatkan Firebase.
10	Menambahkan dan menyelesaikan fungsi verifikasi <i>email</i> dan telepon menggunakan OTP pada halaman Profil, dan fungsi verifikasi telepon menggunakan OTP pada halaman <i>Register</i> . Menambahkan halaman Noesantara untuk kategori alat kesehatan, alat tulis kantor, furnitur, olahan kuliner, dan souvenir.

11	Mengaktifkan fungsi melacak pengiriman dan perhitungan tarif pengiriman pada halaman Kurir Bangsa. Menambahkan fungsi <i>reset password</i> dan tambah <i>password</i> pada halaman Profil. Menambahkan fungsi lupa <i>password</i> pada halaman <i>Login</i> . Mengirimkan <i>email</i> konfirmasi ketika pengguna berhasil mendaftarkan akun. <i>Bug fix</i> halaman Profil, fungsi mengubah informasi alamat, dan verifikasi. <i>Deploy</i> pada server <i>live</i> untuk di-akses secara publik.
12	Mengubah tampilan halaman <i>Login</i> dan <i>Register</i> menyesuaikan dengan desain baru. Menampilkan contoh data ulasan pengguna pada halaman Detail Produk dengan fungsi <i>pagination</i> , <i>show media</i> , dan <i>sort</i> . Mengimplementasikan dan mempresentasikan <i>prototype</i> dari aplikasi <i>chat</i> yang dibuat menggunakan <i>framework</i> Express.js dengan implementasi <i>websocket</i> Socket.IO dengan sistem <i>database</i> MongoDB.
13	Membangun aplikasi <i>chat</i> karyanoesantara.id dengan <i>framework</i> Express.js dan implementasi <i>websocket</i> Socket.IO dengan menggunakan <i>database</i> yang sama dengan karyanoesantara.id dengan sistem <i>database</i> MariaDB MySQL. Aplikasi <i>chat</i> dapat mengirimkan pesan teks. Pesan yang dikirim dan diterima akan bersifat <i>real-time</i> . Aplikasi <i>chat</i> ditampilkan secara <i>iframe</i> .
14	Pembangunan ulang keseluruhan fitur halaman Kategori, yaitu <i>pagination</i> , <i>sorting</i> , jumlah tampil, <i>filter</i> kategori, <i>filter brand</i> , <i>filter</i> lokasi, dan <i>filter</i> harga. Pembangunan ulang dilakukan dengan mengimplementasikan JQuery AJAX untuk setiap fiturnya.
15	Menambahkan fitur pengiriman pesan gambar, dan notifikasi pada aplikasi <i>chat</i> . Menampilkan ongkos kirim dari berbagai macam kurir pengiriman pada halaman Detail Produk. Implementasi <i>skeleton loading</i> pada tampilan ulasan di halaman Detail Produk. <i>Bug fix</i> verifikasi <i>email</i> dan nomor telepon.

16	Menyempurnakan keseluruhan fungsi pada halaman Keranjang seperti pemilihan produk <i>checkout</i> , menghapus produk dipilih, dan memilih semua produk. Menambahkan fitur variasi produk pada halaman Detail Produk dan penyesuaian pada halaman Keranjang. Menambahkan halaman baru Profil Mitra, hanya menampilkan informasi dasar untuk sementara. <i>Security re-check SQL Injection</i> untuk setiap <i>request</i> .
17	Membangun halaman Profil Mitra dengan deskripsi mitra, produk terbaru mitra, dan semua produk mitra secara ringkas pada bagian Beranda. Untuk halaman Kategori, diperbaiki sistem <i>filter</i> secara keseluruhan untuk tampilan <i>desktop</i> seperti <i>filter</i> kategori, <i>brand</i> , lokasi toko, dan harga. Kemudian untuk pengguna <i>mobile device</i> , sistem <i>filter</i> pada halaman kategori dibuat secara khusus. Keseluruhan sistem <i>sort</i> dan <i>filter</i> untuk <i>mobile</i> telah berfungsi dengan baik dengan fitur yang sama dengan tampilan <i>desktop</i> .
18	Optimisasi pengambilan data produk pada halaman Kategori dengan memanfaatkan Axios. Menambahkan fungsi <i>filter</i> ulasan produk pada halaman Kategori untuk tampilan <i>desktop</i> dan <i>mobile</i> . Membangun ulang halaman <i>Checkout</i> dikarenakan program sebelumnya merupakan <i>prototype</i> dan masih memiliki banyak <i>bug</i> yang disebabkan sistem yang berantakan. Pembangunan ulang sudah dilakukan untuk fungsi pemilihan alamat pengguna, daftar produk pesanan pengguna, pemilihan jasa pengiriman, dan jumlah total harga <i>checkout</i> .
19	Membangun sistem <i>checkout</i> pemesanan produk pada halaman <i>Checkout</i> . Metode pembayaran yang sudah berhasil diimplementasikan menggunakan Midtrans API yaitu BCA <i>Virtual Account</i> , Mandiri <i>Virtual Account</i> , BNI <i>Virtual Account</i> , Permata <i>Virtual Account</i> , GoPay, QRIS, Indomaret, dan Alfamart. Metode pembayaran yang telah berhasil diimplementasikan menggunakan Xendit API yaitu OVO, dan LinkAja. Dilakukan penyesuaian pada halaman Pembayaran setiap metode pembayaran. <i>Bug fix</i> halaman Keranjang, fungsi <i>login</i> menggunakan Google dan Facebook, halaman Lupa Password.

20	Membuat <i>API request handler</i> yang diterima dari Midtrans API dan Xendit API untuk masing-masing metode pembayaran untuk menerima informasi penerimaan pembayaran pengguna. Menambahkan sistem metode pembayaran menggunakan Kartu Kredit/Debit dengan memanfaatkan Midtrans API, dan menambahkan metode pembayaran ShopeePay dengan memanfaatkan Xendit API. Melakukan penyesuaian pada tampilan halaman Profil bagian Pesanan Saya sekitar 60%. Sistem pembatalan otomatis untuk pesanan yang telah melewati batas waktu pembayaran.
21	Menambahkan fungsi catatan alamat, dan penggunaan voucher ongkos kirim pada halaman <i>Checkout</i> . Menampilkan informasi ongkos kirim, catatan alamat, catatan pembeli, pada halaman Detail Pesanan. Memperbaiki fitur variasi produk secara keseluruhan.
22	Mengaktifkan fitur pencarian pada halaman Bantuan. Pengiriman otomatis <i>email</i> mengingatkan pembayaran kepada pengguna ketika pengguna memesan produk, <i>email</i> sukses pembayaran kepada pengguna, dan <i>email</i> informasi pemesanan kepada mitra ketika pengguna telah membayar pesanan-nya. Mengubah tampilan sukses <i>login</i> dan <i>register</i> . Mengaktifkan dan penyesuaian sistem <i>login</i> menggunakan Facebook. Implementasi voucher jenis <i>cashback</i> pada halaman <i>Checkout</i> .
23	Perbaiki desain ulang pada halaman Profil bagian Pesanan Saya karena desain yang tidak sesuai dengan sistem pengelompokan pesanan. Menampilkan setiap pesanan dengan pengelompokan sesuai dengan desain terbaru. Fitur <i>show 5 at a time</i> untuk setiap pesanan. Perbaiki <i>login</i> LKPP untuk mengakses halaman Noesantara.
24	Melanjutkan pengembangan halaman Profil Mitra, menambahkan fitur mengikuti mitra tetapi masih hanya berupa data, menampilkan semua produk mitra dengan fungsi <i>sort</i> dan <i>pagination</i> , dan menampilkan semua ulasan produk yang pernah diterima oleh mitra disertai dengan fungsi <i>sort</i> , <i>pagination</i> , dan dengan media, beserta galeri foto pembeli.

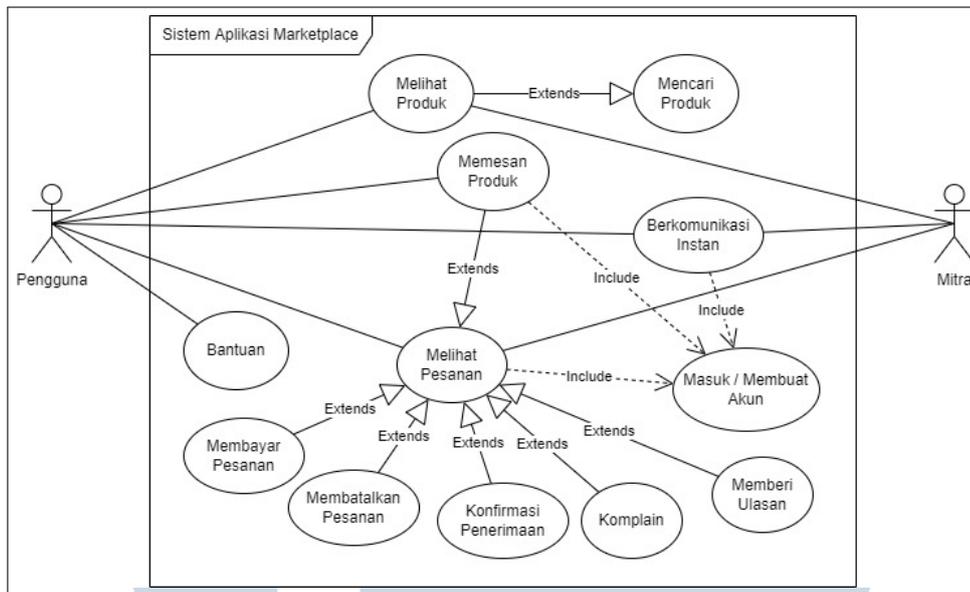
25	Mengaktifkan tombol beli lagi pada halaman Profil bagian Pesanan Saya. Menyempurnakan fitur galeri foto pada halaman Profil Mitra. Menambahkan fitur ganti bahasa pada halaman Profil, namun masih berupa <i>prototype</i> dikarenakan belum semua teks diubah. <i>Bug fix</i> halaman <i>Login</i> , halaman <i>Checkout</i> , dan penggunaan <i>cookie</i> . Pada aplikasi <i>chat</i> , dilakukan peningkatan keamanan pada <i>token</i> dan fitur <i>load more message with scroll</i> .
26	Mengimplementasikan metode pembayaran <i>transfer</i> manual bank BCA dan Mandiri. Pembayaran pengguna akan dikonfirmasi dengan cara pengguna mengirimkan foto bukti pembayarannya dan melakukan <i>upload</i> bukti tersebut pada halaman Pembayaran. Pembuatan Dokumentasi API menggunakan Postman untuk konfirmasi pembayaran manual berhasil. Menambahkan fungsi pembatalan pesanan yang belum dibayar pada halaman Profil bagian Pesanan Saya
27	Mengaktifkan fitur komplain, <i>chat</i> mitra, dan memberi ulasan terhadap produk. Pengguna dapat memberikan ulasan terhadap produk yang telah selesai dipesan dan telah diterima. Pemberian ulasan dapat berupa bintang, atau disertai dengan deskripsi dan gambar-gambar. Pengguna juga dapat mengubah ulasan yang sebelumnya telah diberikan. <i>Deploy</i> ke <i>live server</i> karyanoesantara.id dan juga untuk aplikasi <i>chat</i> yang dijalankan sebagai <i>Node.js server</i> .

Tabel 3.1. Uraian Pelaksanaan Magang Secara Mingguan

Dari Tabel 3.1 dapat dilihat kegiatan per minggu yang dilaksanakan selama praktik kerja magang di PT. Solusi Media Ravel Teknologi dalam mengembangkan *website marketplace* karyanoesantara.id.

3.3.1 Tahapan Perancangan

Dalam perancangan dibutuhkan gambaran umum fungsi penting yang dibutuhkan oleh sebuah aplikasi penjualan. Untuk mempermudah dalam memahaminya, berikut disajikan diagram *Use Case Diagram* Sistem Aplikasi *Marketplace* yang dapat dilihat pada Gambar 3.1.

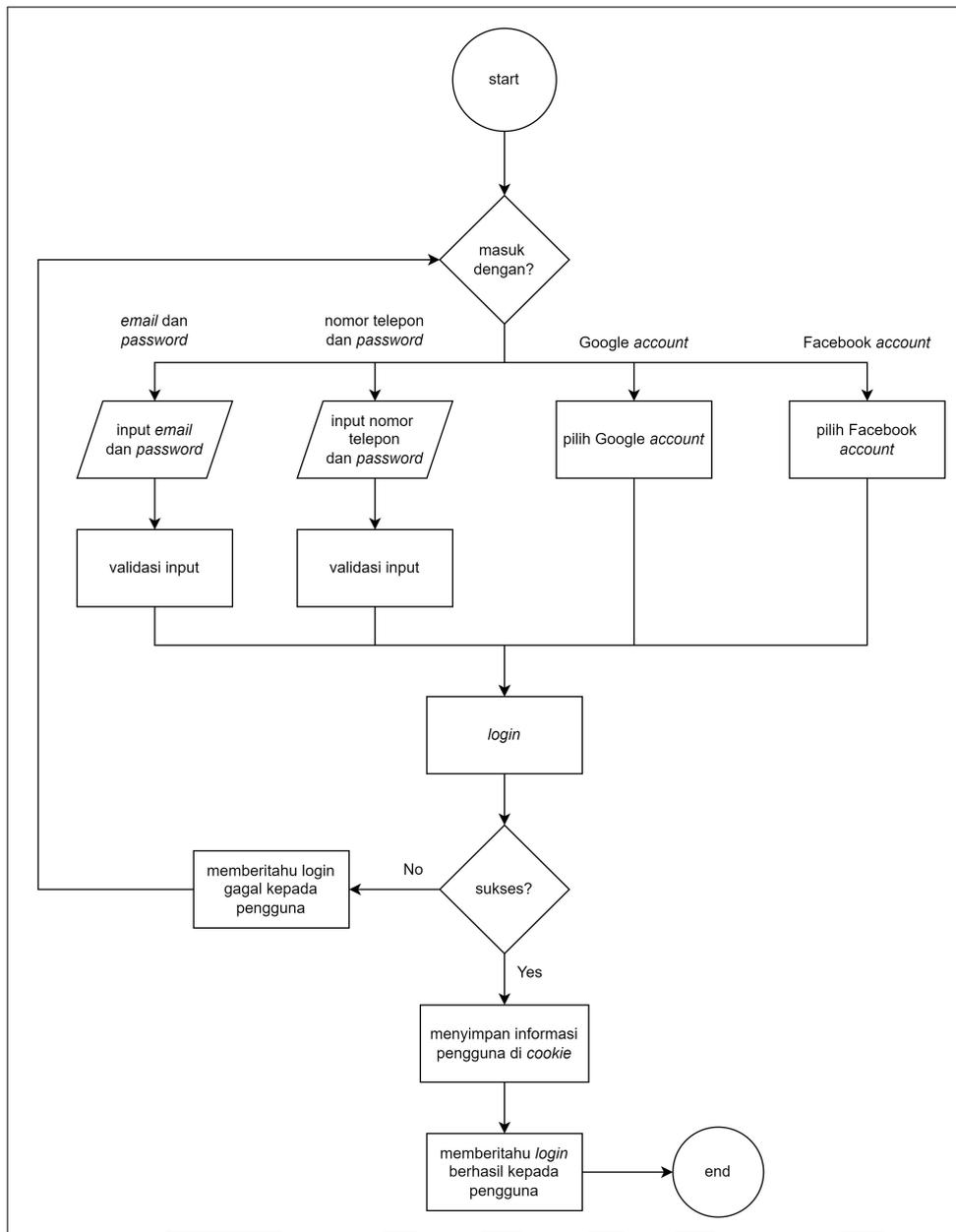


Gambar 3.1. Use Case Diagram Sistem Website Marketplace

Pada Gambar 3.1, baik dari pihak pengguna maupun dari pihak mitra dapat melihat produk-produk yang ada dan mencari produk tertentu. Ketika pengguna ingin melakukan pemesanan produk, pengguna diwajibkan untuk masuk menggunakan akun-nya terlebih dahulu. Setelah pengguna melakukan pemesanan produk, pengguna dapat melihat pesanan yang telah dibuat dan berinteraksi dengan pesanan tersebut seperti membayar pesanan, membatalkan pesanan, konfirmasi penerimaan pesanan, komplain, dan memberi ulasan. Pengguna juga dapat berkomunikasi langsung dengan mitra menggunakan *chat* dan wajib masuk menggunakan akun-nya terlebih dahulu. Apabila pengguna kesulitan dalam menggunakan *website* atau membutuhkan bantuan, pengguna dapat mengakses bantuan secara langsung.

A. Sistem Masuk Akun / Login

Perancangan pada sistem *login* atau masuk menggunakan akun memiliki alur kerja yang dapat dilihat pada *flowchart* Gambar 3.2.



Gambar 3.2. Flowchart Sistem Masuk Akun

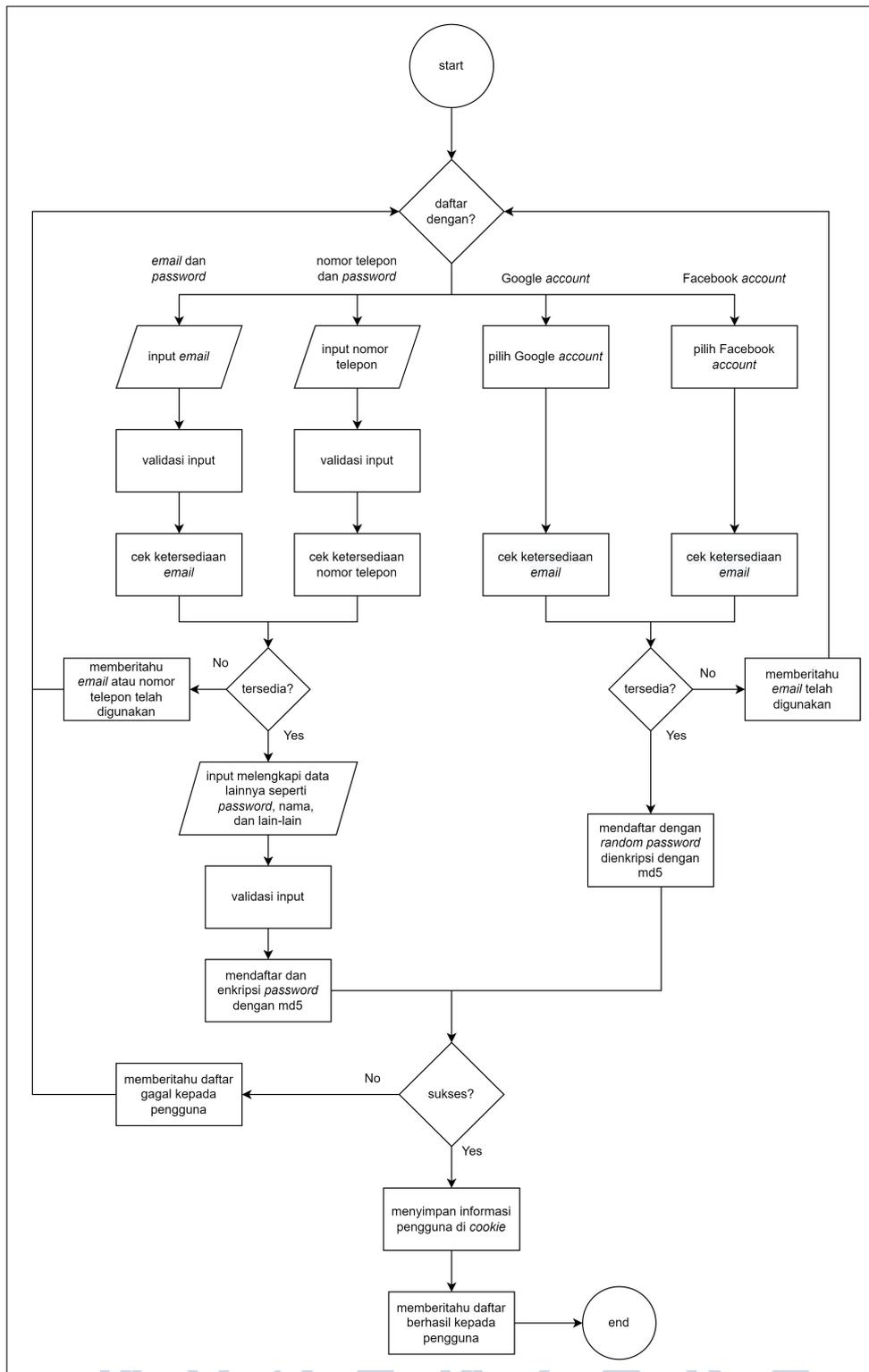
Dari Gambar 3.2, pengguna dapat memilih untuk masuk antara menggunakan *email* dan *password*, nomor telepon dan *password*, *Google account*, dan *Facebook account*. Jika menggunakan *email* dan *password* atau nomor telepon dan *password*, akan dilakukan validasi input dan *login* dilakukan dengan mengenkripsi *password* dengan algoritma md5 dan mencocokkan data *email* dan *password* terenkripsi dengan *database*. Jika menggunakan *Google* atau *Facebook*, akan muncul *pop-up* untuk masuk menggunakan akun *Google*, setelah selesai login

dilakukan dengan pencocokan *email* dengan *database*. Apabila pengguna sukses melakukan *login*, maka informasi pengguna seperti *email* dan nama pengguna akan disimpan di dalam *cookie* pengguna dan memberitahu pengguna bahwa *login* berhasil. Apabila *login* gagal, maka pengguna akan diberitahu bahwa *login* gagal.

B. Sistem Mendaftar Akun / Register

Perancangan pada sistem mendaftar akun memiliki alur kerja yang dapat dilihat pada *flowchart* Gambar 3.3.





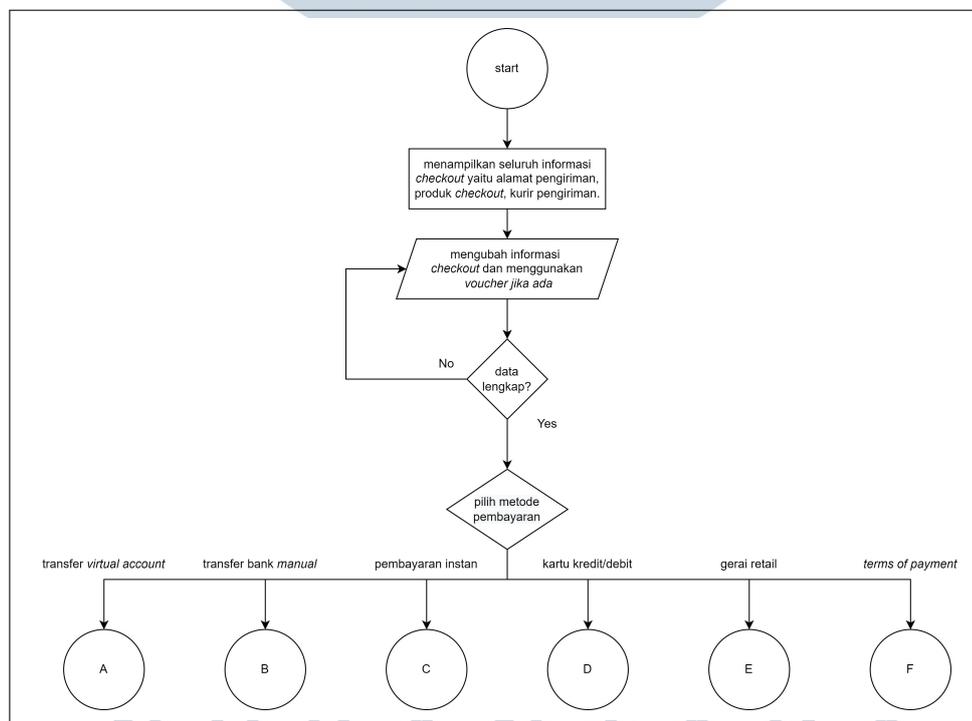
Gambar 3.3. Flowchart Sistem Daftar Akun

Dari Gambar 3.3, pengguna dapat memilih untuk mendaftar antara

menggunakan *email*, nomor telepon, *Google account*, dan *Facebook account*. Apabila pengguna mendaftarkan menggunakan *email* atau nomor telepon, pengguna akan diminta untuk memasukkan data dirinya lagi, sedangkan apabila pengguna mendaftarkan menggunakan *Google account* atau *Facebook account*, pengguna akan langsung terdaftar dengan *generated password*. Ketika pengguna sukses terdaftar, maka informasi pengguna seperti *email* dan nama pengguna akan disimpan di dalam *cookie* pengguna dan memberitahu pengguna bahwa pendaftaran akun berhasil.

C. Sistem Pemesanan Produk

Untuk melakukan pemesanan produk, pengguna dapat melakukan pemesanan dengan mencari terlebih dahulu produk yang diinginkan, kemudian memasukkan produk tersebut ke dalam keranjang. Setelah berhasil dimasukkan ke dalam keranjang, pengguna dapat memilih produk yang diinginkan dari dalam keranjang dan melakukan *checkout* terhadap produk-produk yang dipilih. Perancangan pada sistem *checkout* memiliki alur kerja yang dapat dilihat pada *flowchart* Gambar .

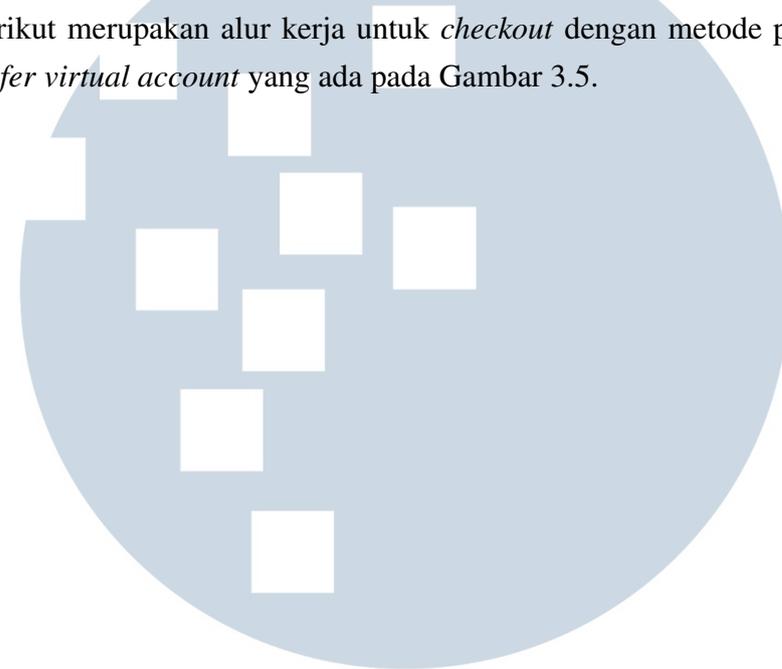


Gambar 3.4. *Flowchart* Sistem *Checkout* Awal

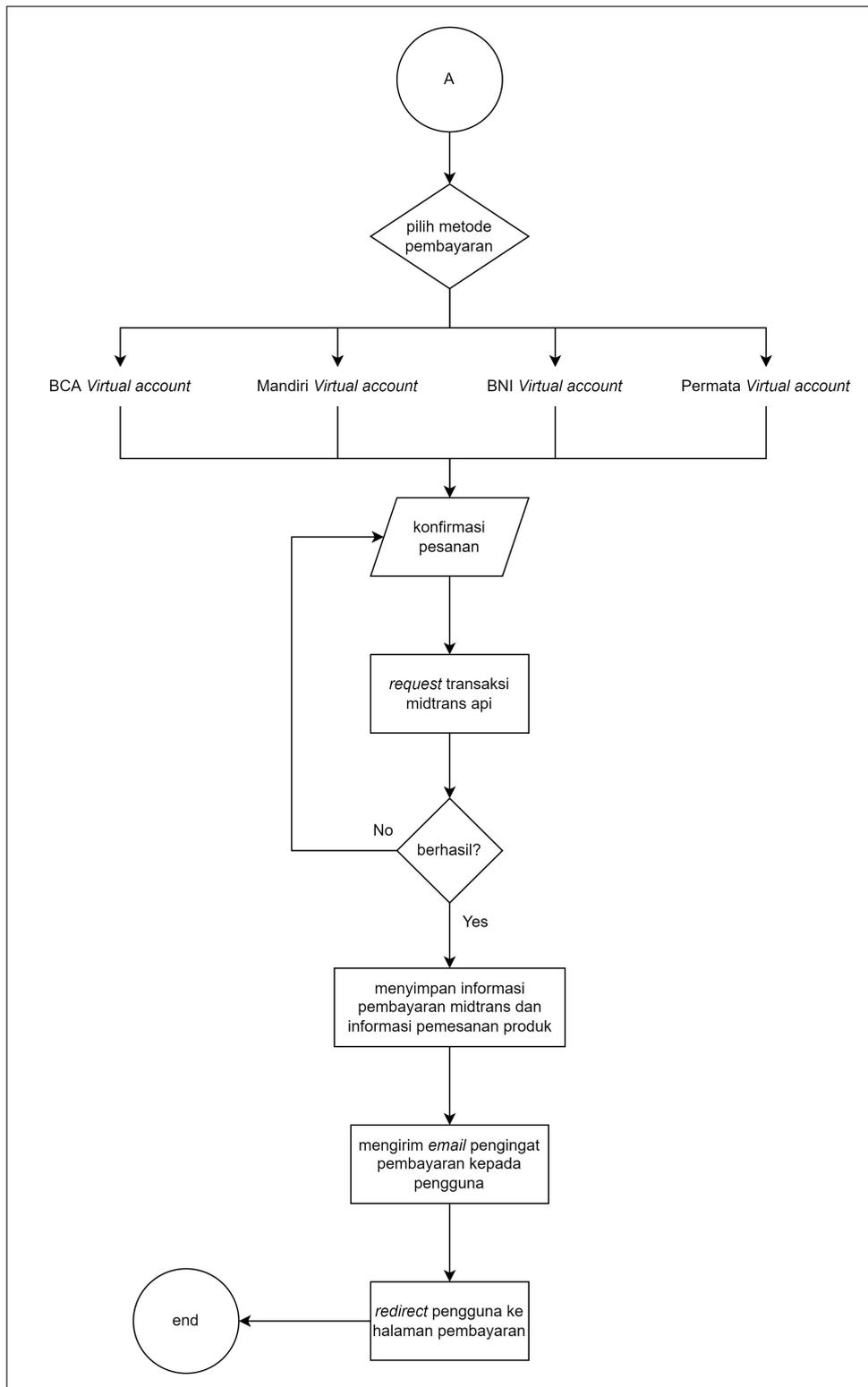
Dari Gambar 3.4, ketika pengguna membuka halaman *Checkout*, akan ditampilkan seluruh informasi *checkout* yaitu alamat pengiriman, produk-produk

yang dipesan, dan kurir pengiriman yang terpilih. Apabila informasi *checkout* sudah lengkap, maka pengguna dapat memilih metode pembayaran. Untuk masing-masing jenis metode pembayaran akan digambarkan pada alur masing-masing.

Berikut merupakan alur kerja untuk *checkout* dengan metode pembayaran jenis *transfer virtual account* yang ada pada Gambar 3.5.



UMMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA

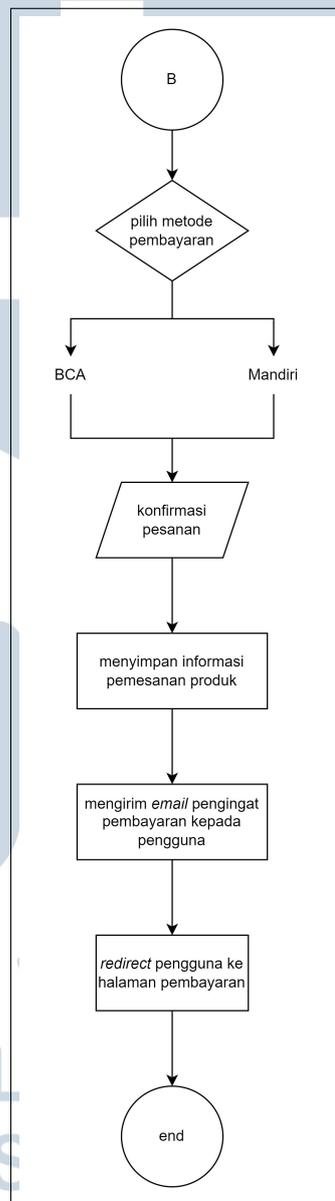


Gambar 3.5. Flowchart Sistem Checkout Transfer Virtual Account

Dari Gambar 3.5, pengguna dapat memilih antara metode pembayaran BCA

Virtual Account, *Mandiri Virtual Account*, *BNI Virtual Account*, dan *Permata Virtual Account*. Setelah pengguna mengkonfirmasi pesanan, akan dilakukan *request* transaksi dari Midtrans API, apabila berhasil, maka akan disimpan informasi pembayaran Midtrans API dan pengguna akan diarahkan menuju halaman Pembayaran.

Berikut merupakan alur kerja untuk *checkout* dengan metode pembayaran jenis *transfer bank manual* yang ada pada Gambar 3.6.

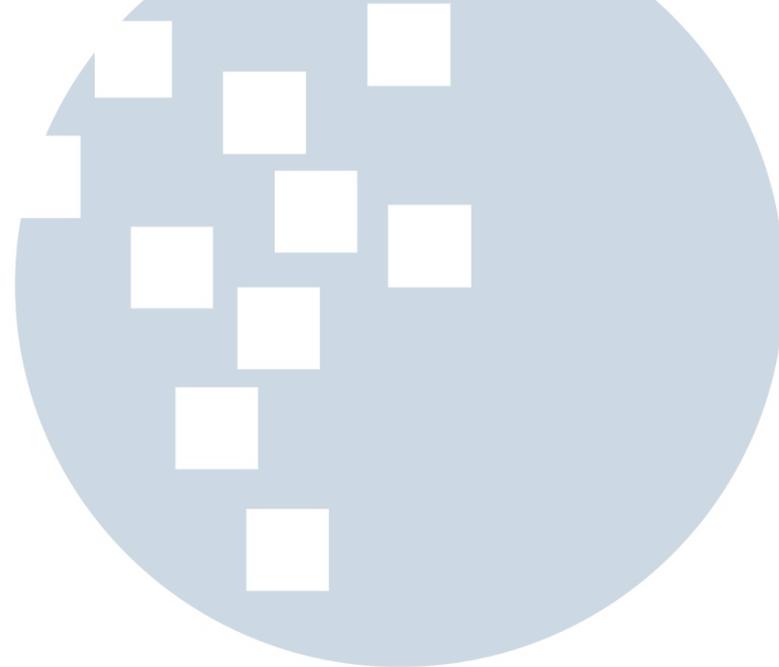


Gambar 3.6. *Flowchart Sistem Checkout Transfer Manual*

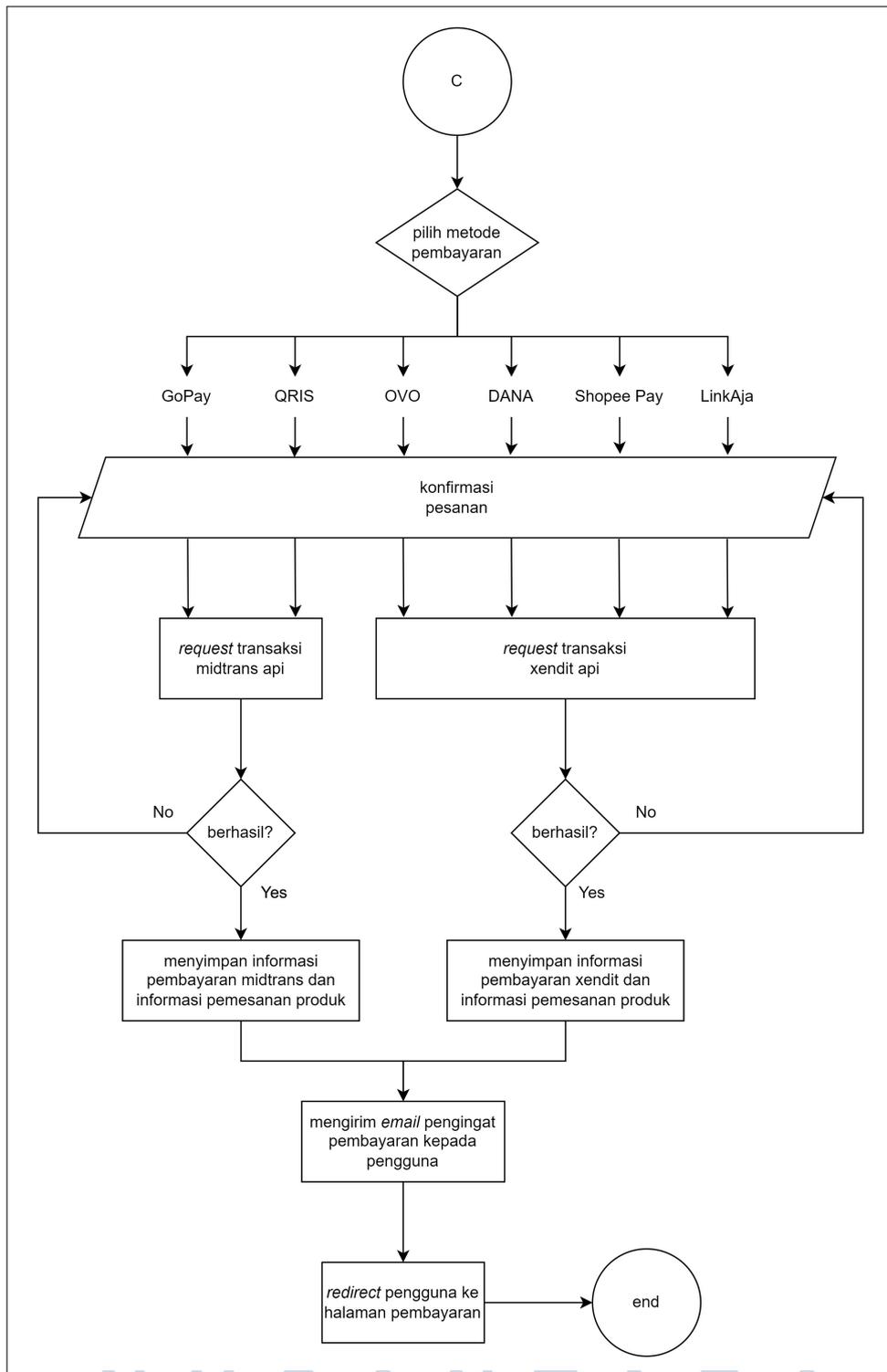
Dari Gambar 3.6, pengguna dapat memilih antara pembayaran melalui bank

BCA atau Mandiri. Setelah melakukan konfirmasi, informasi pemesanan akan disimpan dan pengguna akan diarahkan ke halaman Pembayaran.

Berikut merupakan alur kerja untuk *checkout* dengan metode pembayaran jenis pembayaran instan yang dapat dilihat pada Gambar 3.7.



UMMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA

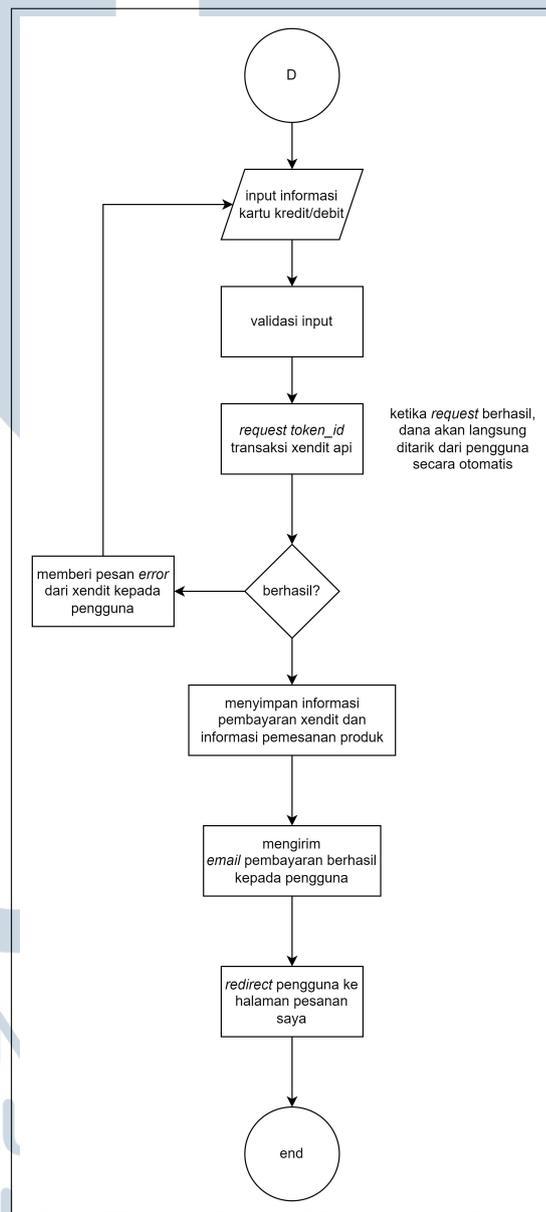


Gambar 3.7. Flowchart Sistem Checkout Pembayaran Instan

Dari Gambar 3.7, pengguna dapat memilih antara pembayaran melalui GoPay, QRIS, OVO, DANA, ShopeePay, dan LinkAja. Untuk metode pembayaran

GoPay dan QRIS, digunakan Midtrans API untuk membantu dalam *generate qr code* nantinya. Untuk metode pembayaran menggunakan OVO, DANA, ShopeePay, dan LinkAja, akan digunakan Xendit API untuk membantu dalam *generate qr code* nantinya.

Berikut merupakan alur kerja untuk *checkout* dengan metode pembayaran jenis kartu kredit atau debit yang dapat dilihat pada Gambar 3.8.

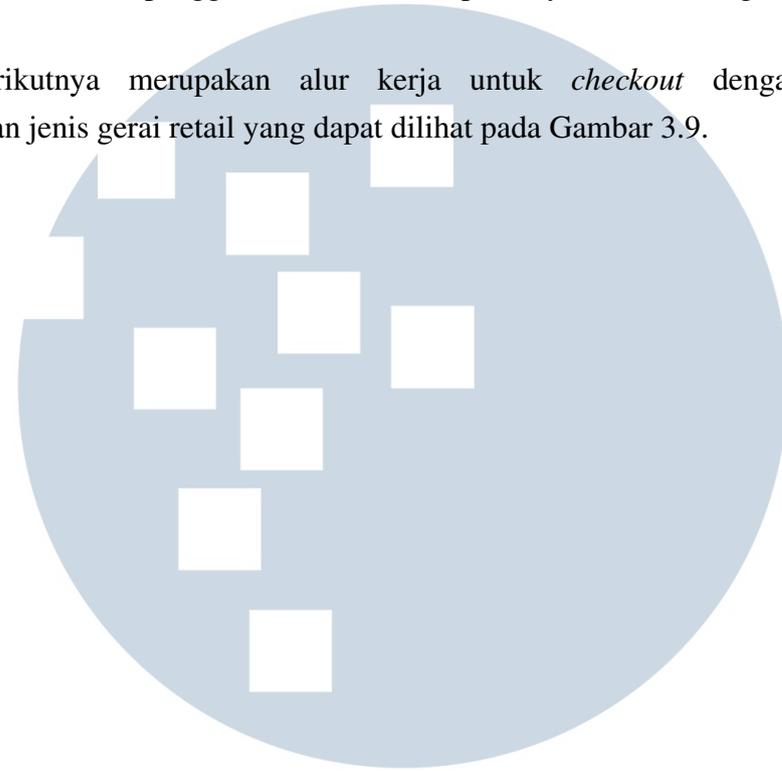


Gambar 3.8. Flowchart Sistem Checkout Kartu Kredit atau Debit

Dari Gambar 3.8, pengguna akan diarahkan untuk memasukkan informasi kartu kredit/debit-nya terlebih dahulu. Kemudian dengan memanfaatkan Xendit

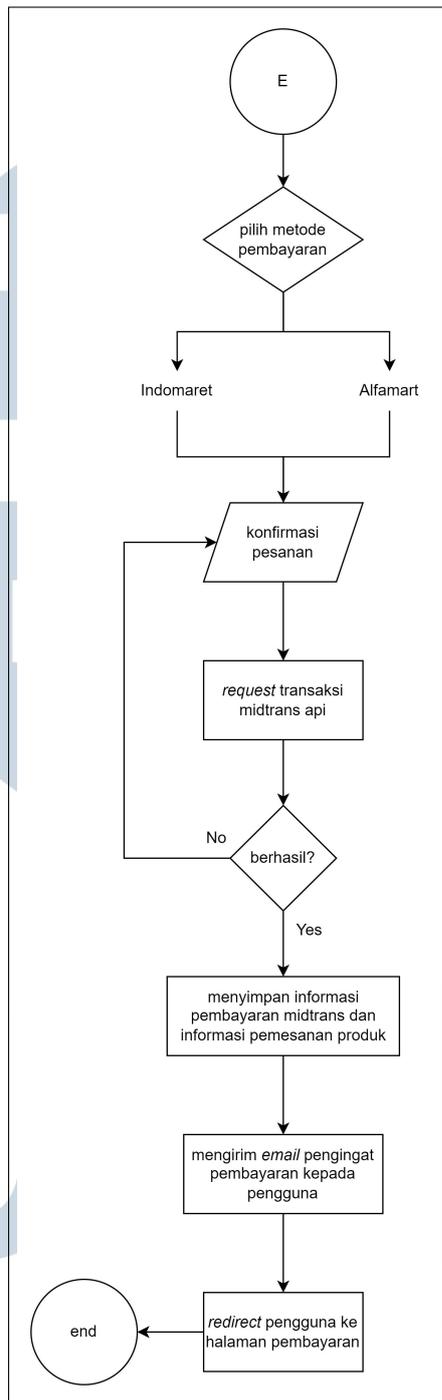
API, kartu kredit/debit pengguna akan di cek dan apabila berlaku, maka dana akan langsung ditarik dari pengguna dan transaksi pembayaran akan langsung berhasil dilakukan.

Berikutnya merupakan alur kerja untuk *checkout* dengan metode pembayaran jenis gerai retail yang dapat dilihat pada Gambar 3.9.



UMMN

UNIVERSITAS
MULTIMEDIA
NUSANTARA

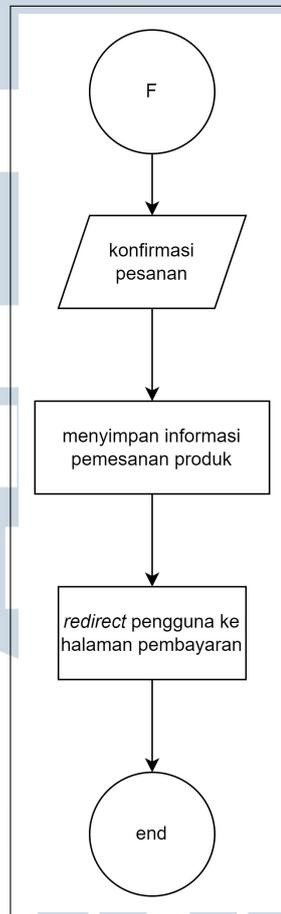


Gambar 3.9. Flowchart Sistem Checkout Gerai Retail

Dari Gambar 3.9, pengguna dapat memilih antara pembayaran melalui Indomaret dan Alfamart. Setelah pengguna melakukan konfirmasi pembayaran, akan dilakukan *request* terhadap Midtrans API untuk membantu dalam pembuatan transaksi. Setelah berhasil, akan disimpan informasi pembayaran Midtrans API dan

informasi pemesanan produk pengguna ke *database*.

Berikut merupakan alur kerja untuk *checkout* dengan metode pembayaran jenis *terms of payment* yang dapat dilihat pada Gambar 3.10.



Gambar 3.10. *Flowchart Sistem Checkout Terms of Payment*

Dari Gambar 3.10, pengguna yang melakukan pembayaran dengan metode pembayaran *terms of payment*, hanya akan dimasukkan informasi pemesanannya ke dalam *database*. Untuk pembayarannya akan dilakukan secara manual nantinya secara bisnis.

Setelah pengguna berhasil melakukan pemesanan produk, pengguna akan dapat melihat setiap informasi pemesanannya pada halaman Profil bagian Pesanan Saya. Pada halaman ini nantinya akan ditampilkan daftar pesanan pengguna dan pengguna dapat berinteraksi dengan pesanan-nya seperti membayar pesanan, membatalkan pesanan, konfirmasi penerimaan pesanan, komplain, dan memberi ulasan setiap produk yang dibeli.

D. Halaman Bantuan Pengguna

Dalam perancangannya, pengguna akan dapat melihat daftar bantuan apa saja yang ditawarkan dan memilih bantuan yang diperlukan. Apabila pengguna tidak dapat menemukan bantuan yang diinginkan, pengguna dapat melakukan pencarian terhadap bantuan yang diperlukan.

E. Sistem Wishlist

Dalam perancangannya, pengguna akan dapat memasukkan produk-produk yang diinginkan untuk dimasukkan ke dalam *wishlist* dengan mengeklik tombol *wishlist*. Produk-produk yang telah dimasukkan ke dalam *wishlist*, nantinya dapat dilihat oleh pengguna pada halaman Profil bagian Wishlist.

F. Halaman Informasi Mitra

Dalam perancangannya, pengguna dapat melihat informasi lengkap mengenai mitra tertentu pada halamannya sendiri. Pada halaman ini nantinya akan ditampilkan produk-produk yang dijual oleh mitra tersebut, dan semua ulasan dari produk-produk yang pernah terjual oleh mitra tersebut.

G. Fitur Chat

Dalam perancangannya, fitur ini akan dibangun secara terpisah dengan *website* utama yaitu *karyanoesantara.id*. Hal ini dikarenakan oleh *website* utama yang dibangun menggunakan PHP Native yang tidak dapat mengimplementasikan penggunaan *websocket* yang merupakan alat paling penting dalam membangun aplikasi *chat*. Aplikasi *chat* ini akan dibangun menggunakan *framework* Express.js yang akan dijalankan dibawah Node.js *server*. *Framework* Express.js dipilih karena dapat mengimplementasikan salah satu *library websocket* yaitu Socket.IO. Aplikasi *chat* akan digunakan untuk berkomunikasi antara pengguna dengan mitra. Pengguna dan mitra dapat mengirimkan pesan berupa teks maupun gambar terhadap satu sama lain.

3.3.2 Tahapan Implementasi

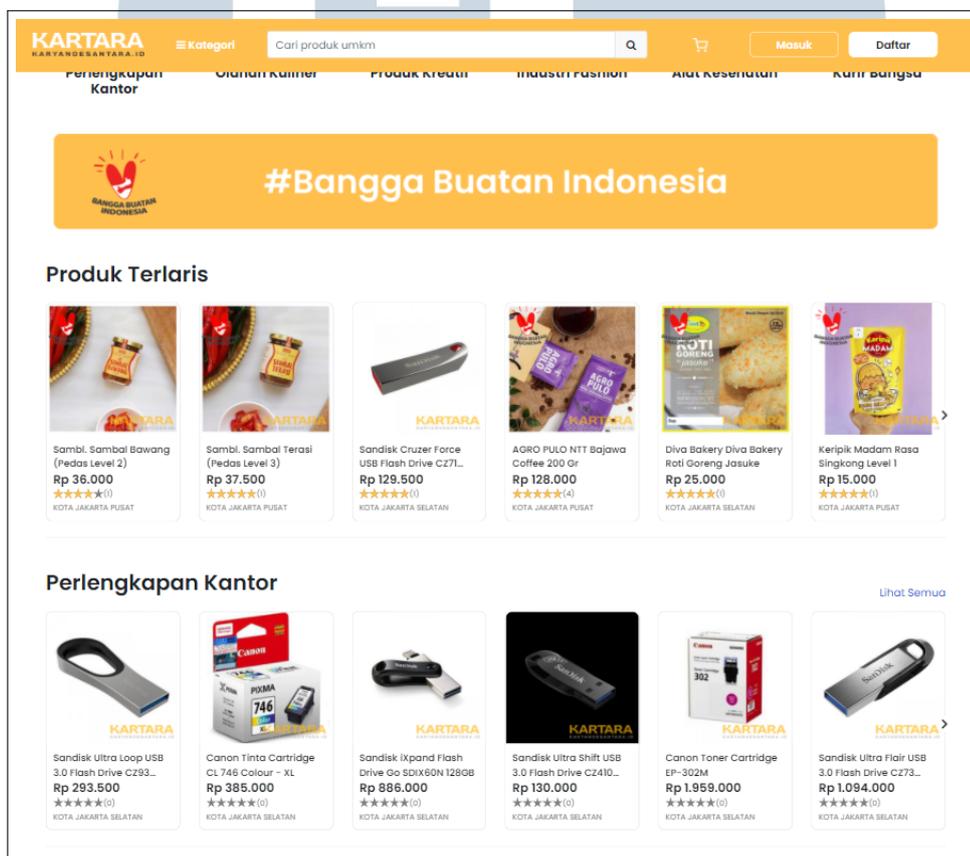
Dalam implementasinya, *website* yang digunakan oleh pengguna untuk melakukan pencarian produk, pemesanan, dan transaksi jual beli produk, dilakukan di <https://karyanoesantara.id>. Sedangkan *website* yang digunakan oleh mitra untuk memperjualbelikan dan manajemen produknya, dilakukan di <https://mitra>.

karyanoesantara.id. Implementasi di bawah ini akan dijelaskan berdasarkan perancangan yang telah dibuat.

A. Melihat Produk

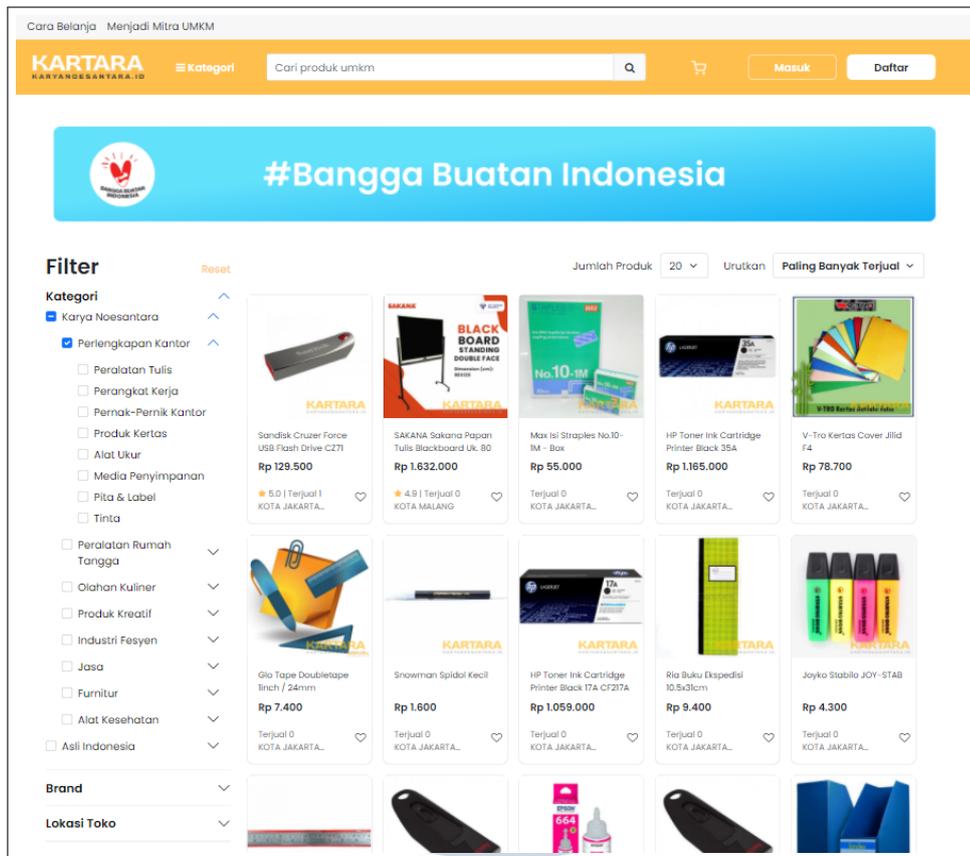
A.1 Melihat Berdasarkan Kategori dan *Filter*

Pengguna dapat melihat produk berdasarkan kategorinya dan *filter* yang telah dimasukkan oleh pengguna. Pada Gambar 3.11 di bawah ini merupakan tampilan produk pada halaman beranda.



Gambar 3.11. Melihat Produk di Beranda

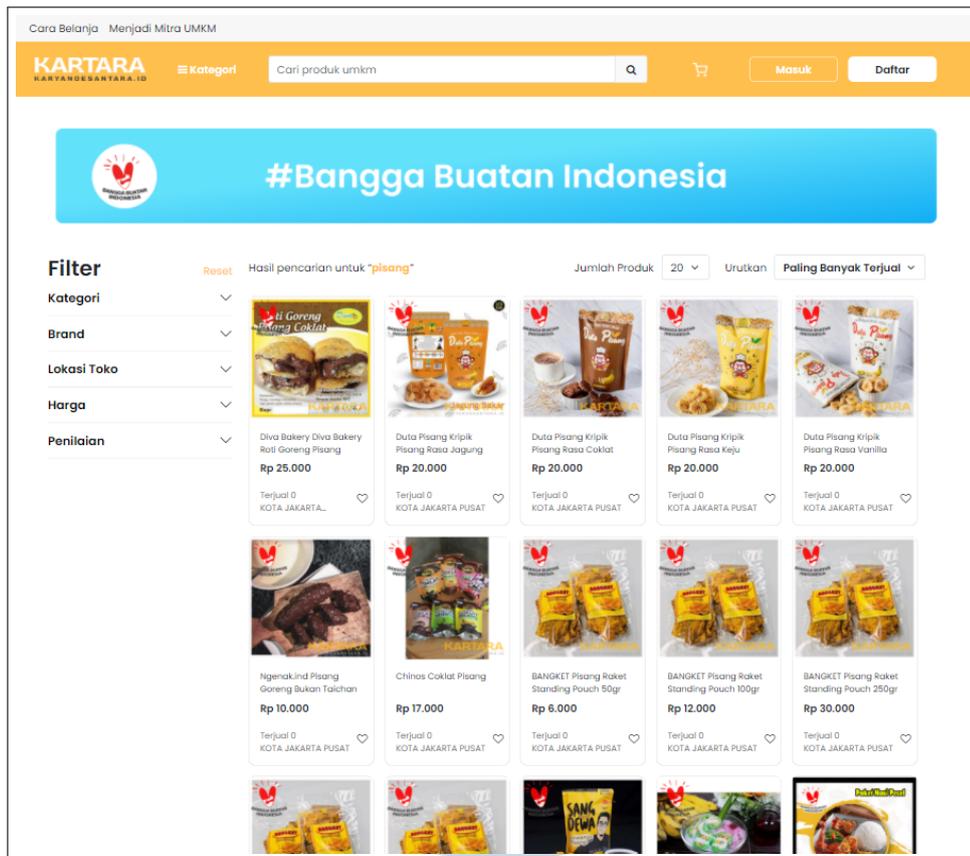
Pada Gambar 3.11, terdapat 10 produk yang ditampilkan secara acak sesuai dengan kategorinya masing-masing beserta penilaiannya. Berikutnya pada Gambar 3.12 di bawah ini merupakan tampilan produk pada saat pengguna membuka kategori tertentu.



Gambar 3.12. Melihat Produk Kategori Tertentu

Pada Gambar 3.12, merupakan tampilan ketika pengguna memilih kategori tertentu untuk melihat produk-produk yang tersedia. Produk-produk tersebut ditampilkan beserta penilaian-nya berdasarkan *filter* yang terdapat di bagian kiri, dan paginasi. Berikutnya pada Gambar 3.13 di bawah ini merupakan tampilan produk ketika pengguna melakukan pencarian.

UNIVERSITAS
MULTIMEDIA
NUSANTARA



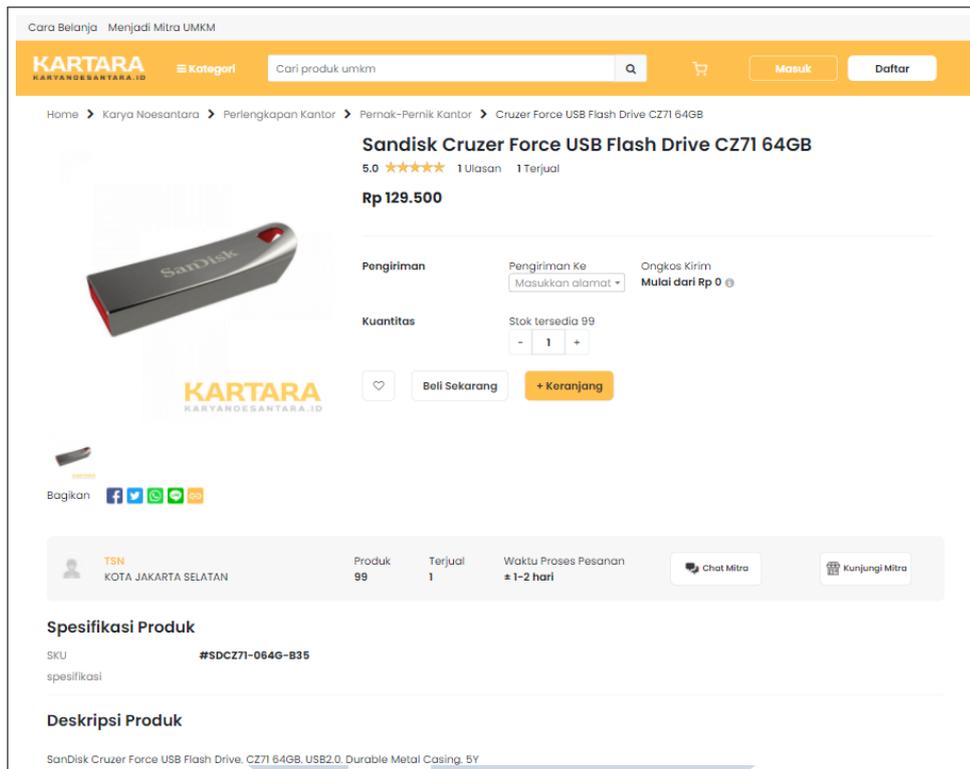
Gambar 3.13. Melihat Produk dengan Pencarian

Pada Gambar 3.13, produk-produk ditampilkan beserta penilaiannya berdasarkan kata kunci pencarian, *filter* yang terdapat di bagian kiri, dan paginasi.

A.2 Melihat Detail Produk

Pengguna dapat melihat informasi detail mengenai produk tertentu. Berikut merupakan tampilan detail produk seperti yang terlihat pada Gambar 3.14.

UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.14. Melihat Detail Produk

Pada Gambar 3.14, informasi produk ditampilkan dengan lengkap beserta penilaian produk tersebut.

B. Masuk dan Membuat Akun

Pengguna dapat masuk atau membuat akun baru dengan berbagai metode. Metode yang dapat digunakan oleh pengguna yaitu sebagai berikut:

1. Menggunakan *email* dan *password*.
2. Menggunakan nomor telepon dan *password*.
3. Menggunakan akun Google.
4. Menggunakan akun Facebook.

Gambar 3.15 di bawah ini merupakan tampilan Masuk Akun.

KARTARA
KARYANOEANTARA.ID

Masuk

Nomor HP atau Email

Misal: 08123456789, atau nama@email.com

Password

[Lupa Password?](#)

Masuk

ATAU

Facebook

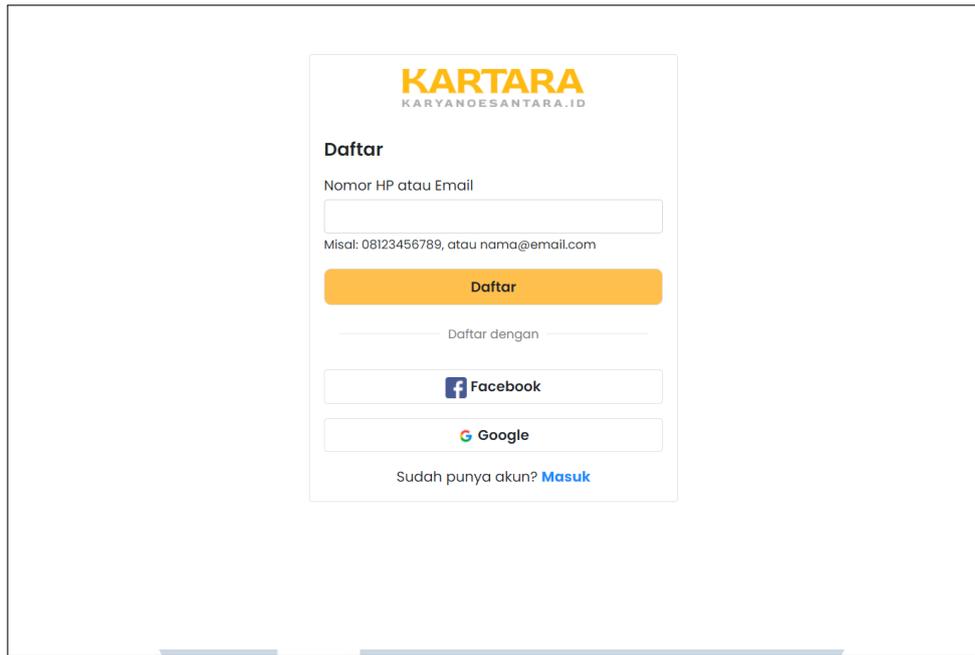
Google

Belum punya akun? [Daftar Sekarang](#)

Gambar 3.15. Tampilan Masuk Akun

Dari Gambar 3.15, pengguna dapat masuk menggunakan *email* dan *password* dengan memasukkan *email* dan *password* sesuai inputnya. Pengguna juga dapat masuk menggunakan Google dengan mengeklik tombol Google atau masuk menggunakan Facebook dengan mengeklik tombol Facebook. Gambar 3.16 dibawah ini merupakan tampilan pertama untuk membuat akun.

UMMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.16. Tampilan Pertama Membuat Akun

Gambar 3.16 di atas merupakan tampilan pertama ketika pengguna ingin mendaftarkan akun barunya. Berikutnya merupakan tampilan berikutnya ketika pengguna telah memasukkan alamat email yang ingin digunakan.

UMMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA

KARTARA
KARYANONESANTARA.ID

Daftar

Nomor HP atau Email
danielwj@gmail.com
Misal: 08123456789, atau nama@email.com

Nama Lengkap

Laki-laki Perempuan

Password

Konfirmasi Password

Perorangan Instansi Perusahaan

Saya telah membaca dan menyetujui [Aturan Penggunaan dan Kebijakan Privasi KARTARA](#)

Daftar

Daftar dengan

Facebook

Google

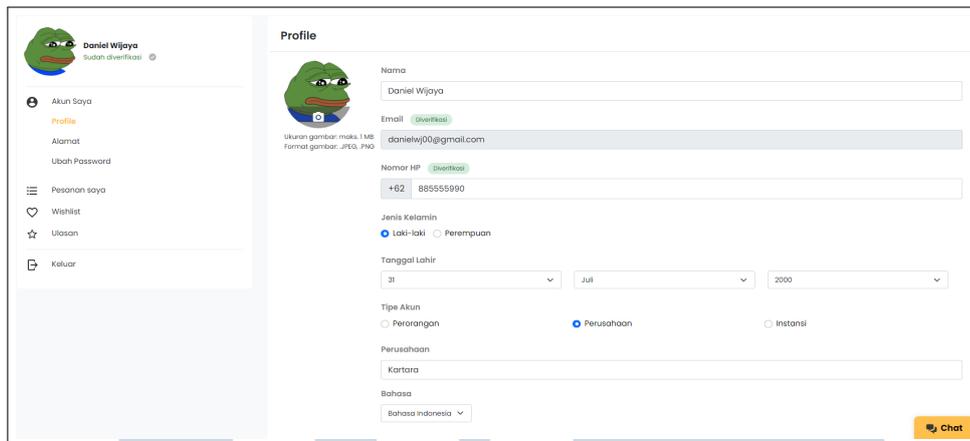
Sudah punya akun? [Masuk](#)

Gambar 3.17. Tampilan Kedua Membuat Akun

Pada Gambar 3.17, pengguna diberikan pilihan untuk membuat akun baru dengan *email* dan *password*, atau menggunakan akun Google dan Facebook.

Untuk implementasi masuk dan membuat akun menggunakan *email* dan *password*, maupun menggunakan Google dan Facebook, informasi *email* pengguna akan disimpan di dalam *database*. Sehingga walaupun pengguna masuk menggunakan *email* dan *password*, pengguna dapat masuk menggunakan Google atau Facebook dengan *email* yang sama dan akan tetap masuk ke dalam akun yang sama. Dalam implementasi masuk menggunakan Google dan Facebook, dimanfaatkan fitur dari Firebase untuk mempermudah dalam mengimplementasikannya.

Setelah pengguna berhasil memiliki akun atau masuk menggunakan akun yang sudah ada, pengguna dapat melihat informasi mengenai data profilnya seperti yang ditampilkan pada Gambar 3.18.



Gambar 3.18. Tampilan Informasi Pengguna

Pada Gambar 3.18, terdapat informasi dasar pengguna seperti nama, *email*, nomor telepon, jenis kelamin, dan tanggal lahir, disertai dengan tipe akunnya dan pemilihan bahasanya.

C. Verifikasi Email dan Nomor Telepon

Pengguna dapat memverifikasi *email* dan nomor telepon yang digunakan pada halaman Profil. Untuk melakukan verifikasi *email*, pengguna akan dikirimkan kode OTP lewat email yang digunakan untuk verifikasi. Pengguna kemudian akan menggunakan kode OTP tersebut untuk melakukan verifikasi. Pengiriman email ini menggunakan alat pengiriman *email* bawaan PHP. Untuk melakukan verifikasi nomor telepon, pengguna akan dikirimkan kode OTP ke nomor telepon yang telah dimasukkan untuk diverifikasi. Pengguna kemudian akan diminta untuk memasukkan kode OTP tersebut untuk melakukan verifikasi. Berikut merupakan contoh pengguna yang belum melakukan verifikasi nomor telepon yang dapat dilihat pada Gambar 3.19.

UNIVERSITAS
MULTIMEDIA
NUSANTARA

The image shows a user profile form with the following fields and options:

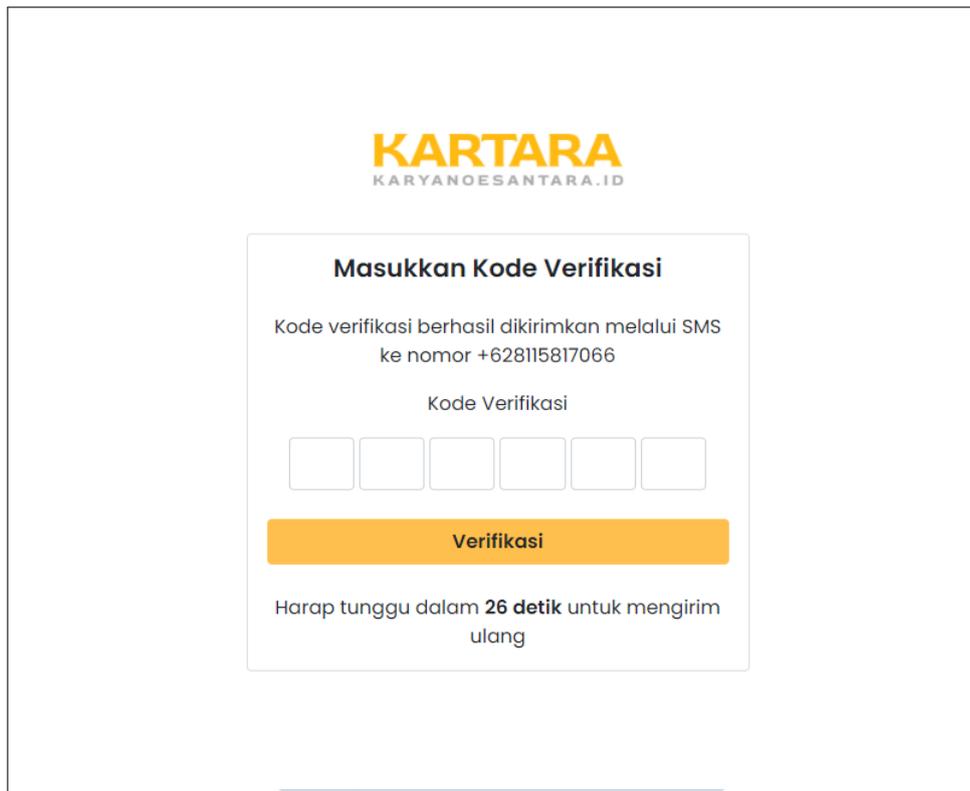
- Nama:** Daniel Wijaya
- Email:** danielwij33@gmail.com (Status: belum diverifikasi)
- Nomor HP:** +62 8115817066 (Status: belum diverifikasi)
- Jenis Kelamin:** Laki-laki Perempuan
- Tanggal Lahir:** Hari, Bulan, Tahun (dropdown menus)
- Tipe Akun:** Perorangan Perusahaan Instansi
- Bahasa:** Bahasa Indonesia (dropdown menu)

Buttons: Simpan (bottom left), Chat (bottom right)

Gambar 3.19. Tampilan Pengguna Belum Verifikasi *Email* dan Nomor Telepon

Dapat dilihat pada Gambar 3.19, pengguna belum melakukan verifikasi terhadap *email* dan nomor teleponnya. Ketika pengguna mengklik tombol verifikasi pada bagian sebelah kanan nomor telepon pengguna, pengguna akan diarahkan pada halaman pengisian kode OTP. Berikut merupakan tampilan ketika pengguna diminta untuk memasukkan kode OTP yang dapat dilihat pada Gambar 3.20.

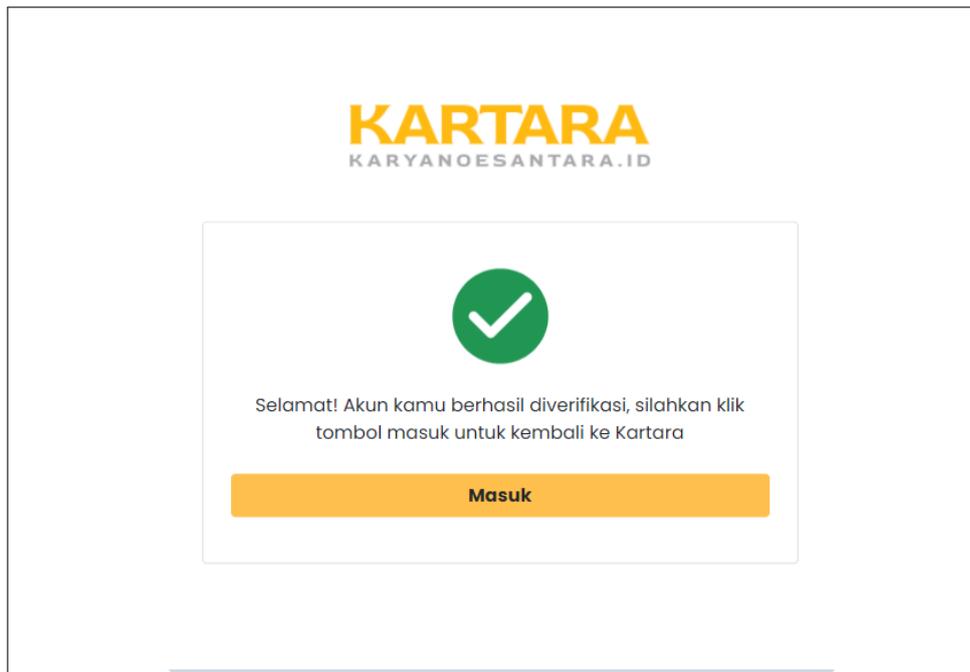
UMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.20. Tampilan Pengguna Diminta Memasukkan Kode Verifikasi OTP

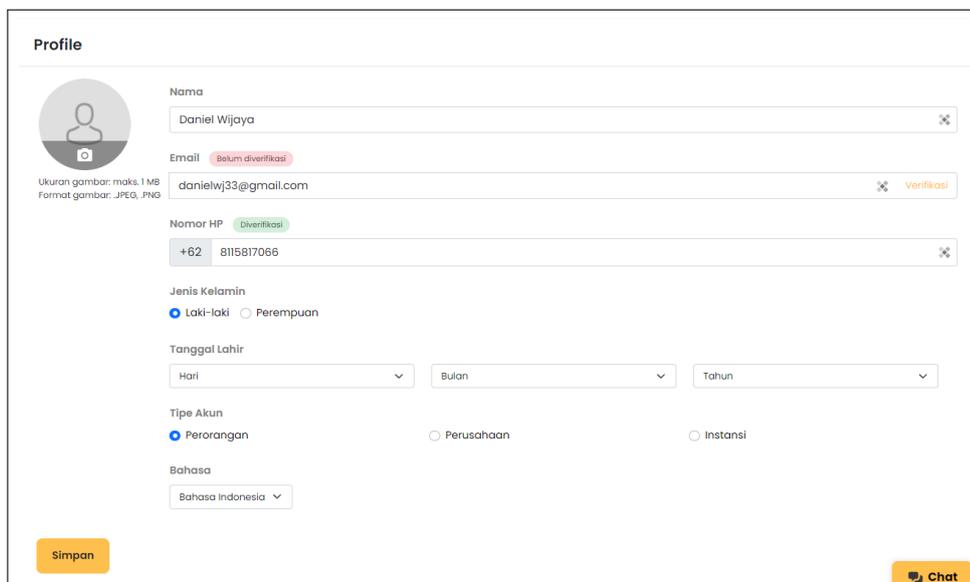
Dapat dilihat pada Gambar 3.20, pengguna diminta untuk memasukkan kode verifikasi OTP yang telah dikirimkan ke nomor telepon pengguna. Pengiriman kode OTP ini diimplementasikan dengan memanfaatkan Firebase. Pengguna dapat meminta pengiriman ulang kode OTP setelah 30 detik. Setelah pengguna memasukkan kode OTP yang benar, maka akan muncul tampilan berhasil seperti Gambar 3.21.

U M M N
UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.21. Tampilan Pengguna Berhasil Verifikasi

Pada Gambar 3.21, pengguna berhasil melakukan verifikasi dan data telah berhasil tersimpan. Pengguna dapat melihat tanda bahwa nomor telepon pengguna telah berhasil diverifikasi pada Gambar 3.22.



Gambar 3.22. Tampilan Pengguna Berhasil Verifikasi Nomor Telepon

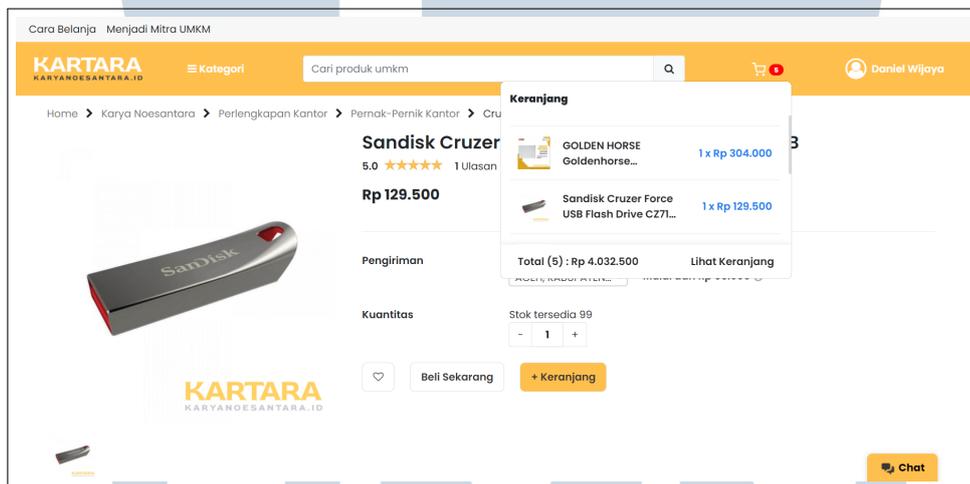
Dapat dilihat pada Gambar 3.22, pengguna telah berhasil di-verifikasi nomor telepon-nya.

D. Memesan Produk

Untuk memesan produk, pengguna harus masuk ke dalam akunnya terlebih dahulu. Jika pengguna belum masuk, maka pengguna akan diarahkan untuk masuk akun terlebih dahulu. Terdapat tahapan untuk melakukan pemesanan produk, dibawah ini merupakan tahap-tahapan yang diperlukan untuk memesan produk.

D.1 Memasukkan Produk ke dalam Keranjang

Untuk memasukkan produk ke dalam keranjang, pengguna dapat membuat detail produk, kemudian mengklik tombol "+Keranjang". Jika berhasil maka isi keranjang akan ditampilkan seperti Gambar 3.23 di bawah.

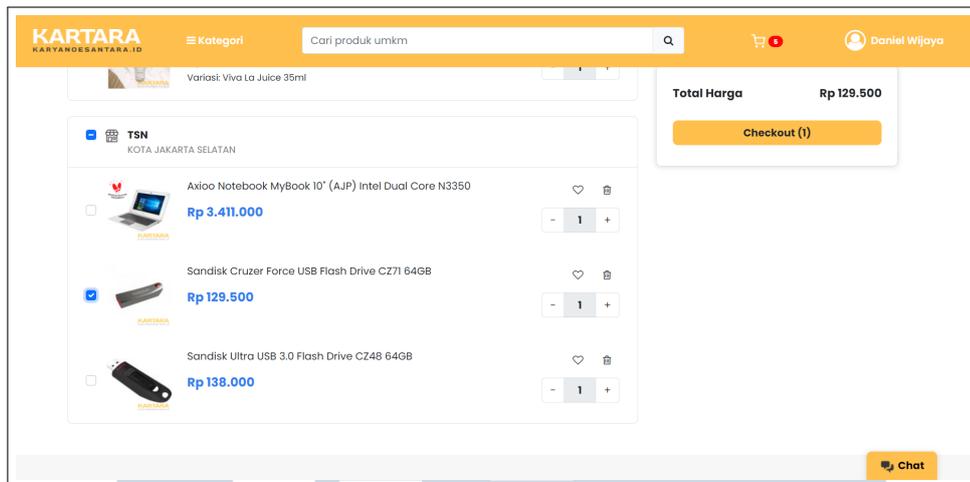


Gambar 3.23. Tampilan Produk di Keranjang

Dari Gambar 3.23, produk telah berhasil dimasukkan ke dalam keranjang.

D.2 Memilih Produk dalam Keranjang

Setelah produk ditambahkan ke dalam keranjang, pengguna akan memilih produk yang terdapat di dalam keranjang untuk dibawa *checkout*. Untuk memilihnya seperti pada Gambar 3.24 berikut.

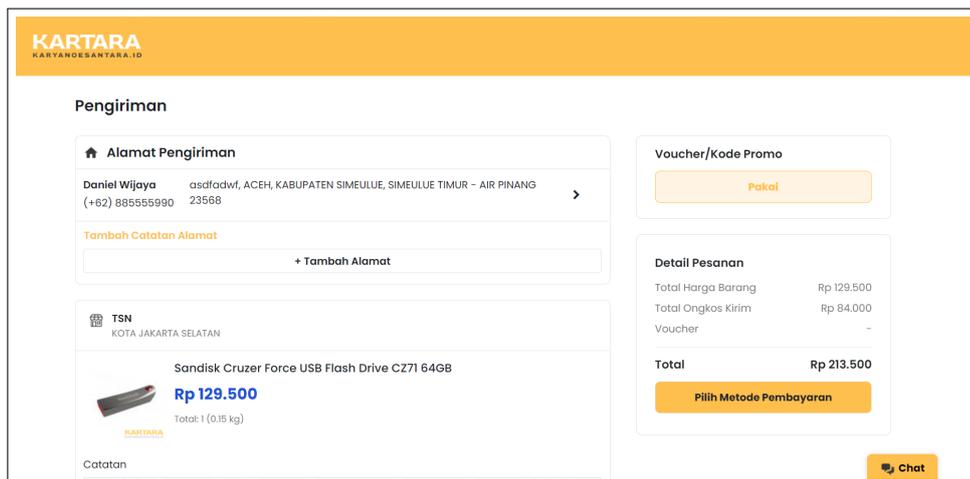


Gambar 3.24. Memilih Produk dalam Keranjang

Seperti pada Gambar 3.24, produk di tandai dengan *checkbox* bahwa produk tersebut akan dipesan.

D.3 Melengkapi Informasi Pemesanan

Setelah memilih produk untuk dipesan, pengguna diwajibkan untuk memenuhi informasi yang diperlukan agar dapat melakukan pemesanan. Tampilan pemesanan yang lengkap seperti Gambar 3.25 berikut.

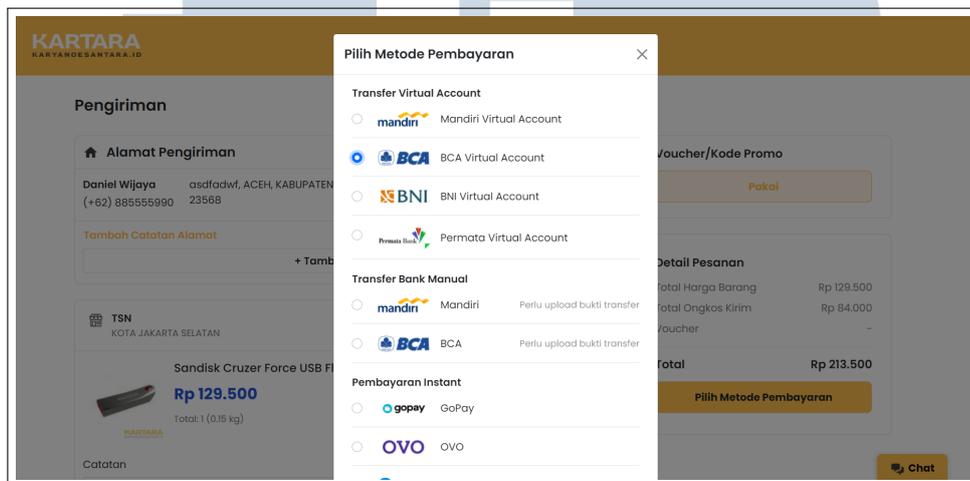


Gambar 3.25. Informasi Pesanan

Seperti pada Gambar 3.25, informasi yang dibutuhkan seperti alamat pengiriman yang digunakan, kurir pengiriman yang digunakan, dan catatan lainnya apabila dibutuhkan.

D.4 Memilih Metode Pembayaran

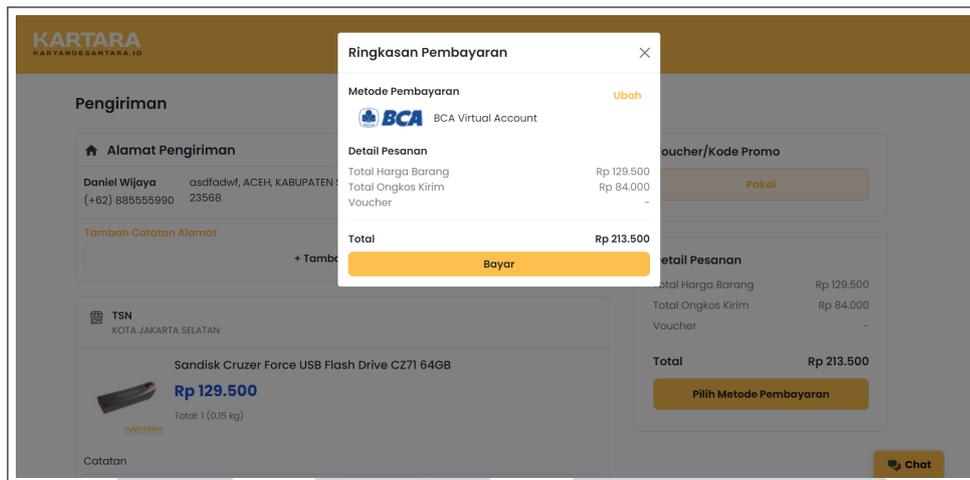
Setelah memenuhi informasi yang diperlukan untuk melakukan pemesanan, pengguna diarahkan untuk memilih metode pembayaran. Pengguna tidak akan dapat mengubah metode pembayarannya ketika pesanan telah dimasukkan, sehingga penting untuk memastikan metode pembayaran yang telah dipilih sudah benar. Berikut merupakan tampilan ketika pengguna diminta untuk memilih metode pembayaran.



Gambar 3.26. Memilih Metode Pembayaran

Dari Gambar 3.26, pengguna akan memilih salah satu dari berbagai macam metode pembayaran yang dapat digunakan. Setelah metode pembayaran dipilih, akan muncul ringkasan pembayaran yang akan dilakukan oleh pengguna seperti Gambar 3.27.

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A

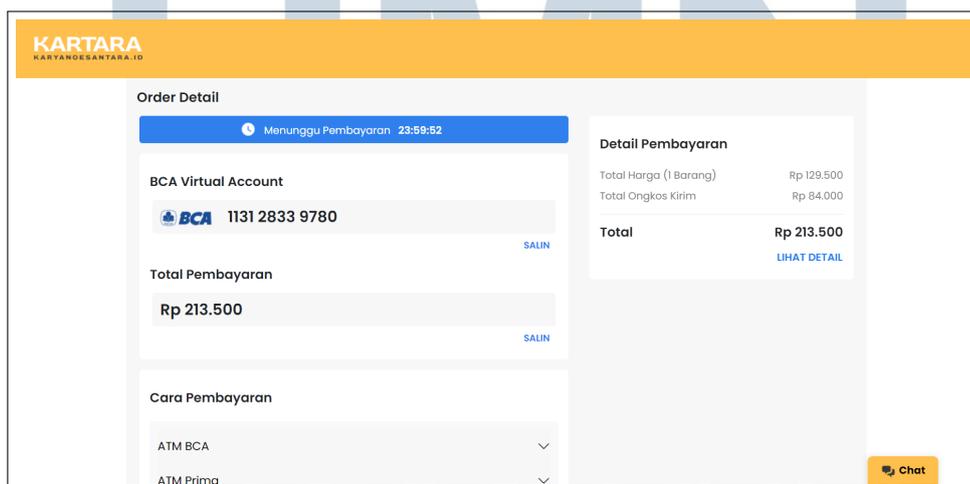


Gambar 3.27. Ringkasan Pembayaran

Dari Gambar 3.27, ketika pengguna mengeklik bayar, maka pesanan pengguna akan diproses langsung sesuai dengan metode pembayaran yang telah dipilih.

D.5 Pemesanan Berhasil

Setelah pesanan diproses, pengguna akan diminta untuk membayar pesannya. Pada tahap ini, pemesanan telah berhasil dilakukan. Berikut tampilan ketika pengguna diminta untuk membayar metode pembayaran transfer akun virtual BCA.



Gambar 3.28. Pemesanan Berhasil

Dari Gambar 3.28, terdapat nomor virtual yang digunakan untuk menerima pembayaran dari pengguna beserta waktu kedaluwarsa pembayaran tersebut.

E. Membayar Pesanan

Metode pembayaran yang sangat beragam, menyebabkan terdapat implementasi yang berbeda-beda untuk beberapa metode pembayaran. Secara garis besar terdapat 4 macam implementasi yang digunakan, yakni:

1. Penggunaan Midtrans API.
2. Penggunaan Xendit API.
3. Pengecekan Manual Bukti Transfer.
4. Pengecekan Manual.

Berikut disajikan pada Tabel 3.2 untuk memperjelas penggunaan implementasi pada masing-masing metode pembayaran.



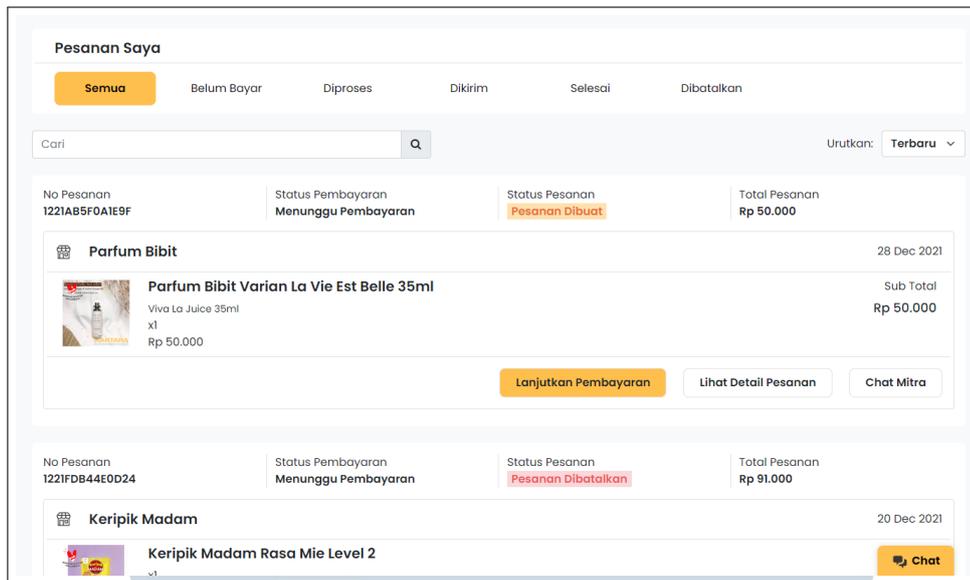
Metode Pembayaran	Implementasi
Transfer Virtual Account	
Mandiri <i>Virtual Account</i>	Midtrans API
BCA <i>Virtual Account</i>	Midtrans API
BNI <i>Virtual Account</i>	Midtrans API
Permata <i>Virtual Account</i>	Midtrans API
Transfer Bank Manual	
Mandiri	Pengecekan Manual Bukti Transfer
BCA	Pengecekan Manual Bukti Transfer
Pembayaran Instan	
GoPay	Midtrans API
OVO	Xendit API
DANA	Xendit API
ShopeePay	Xendit API
LinkAja	Xendit API
QRIS	Midtrans API
Kartu Kredit/Debit	
Kartu Kredit/Kartu Debit	Xendit API
Gerai Retail	
Indomaret	Midtrans API
Alfamart	Midtrans API
Lainnya	
Terms of Payment (TOP)	Pengecekan Manual

Tabel 3.2. Implementasi Metode Pembayaran

Dari Tabel 3.2, terlihat dengan jelas setiap metode pembayaran menggunakan implementasi apa saja.

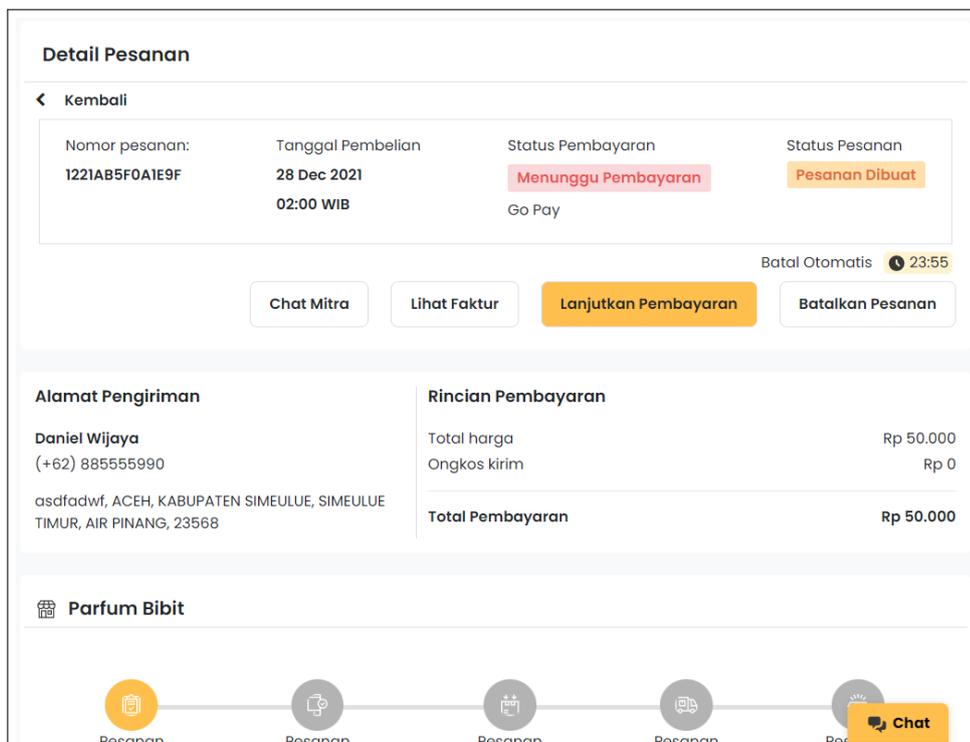
E. Melihat Pesanan

Untuk mempermudah pengguna dalam melihat pesanan, diberikan tampilan pesanan pengguna dengan beberapa interaksi yang dapat dilakukan oleh pengguna. Gambar 3.29 berikut merupakan tampilan dari halaman pesananan pengguna.



Gambar 3.29. Melihat Pesanan Pengguna

Dari Gambar 3.29, pengguna dapat melihat pesannya sesuai dengan statusnya, dan berinteraksi dengan pesannya. Salah satu interaksinya adalah melihat detail dari pesanan tersebut. Gambar berikut merupakan sebagian tampilan dari salah satu detail pesannya.

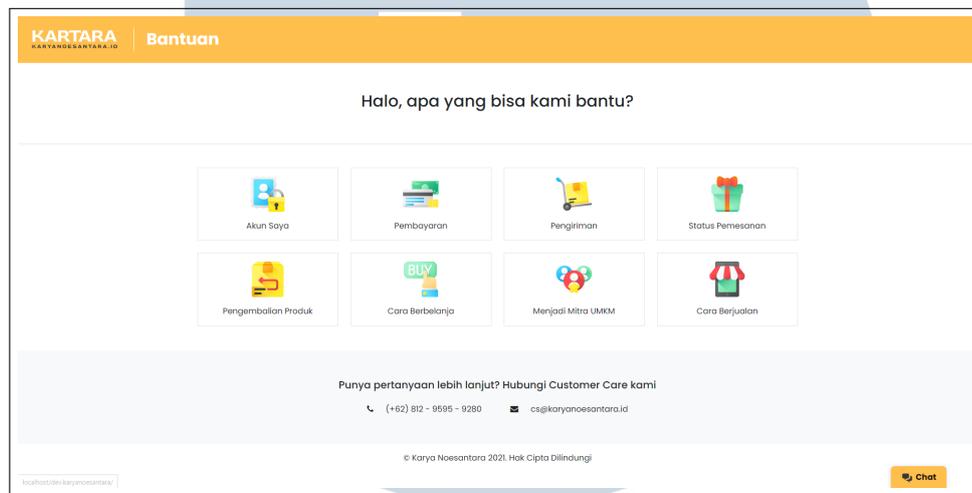


Gambar 3.30. Melihat Detail Pesanan Pengguna

Dari Gambar 3.30, ditampilkan sebagian tampilan dari detail pesanan yang telah dipesan oleh pengguna sebelumnya.

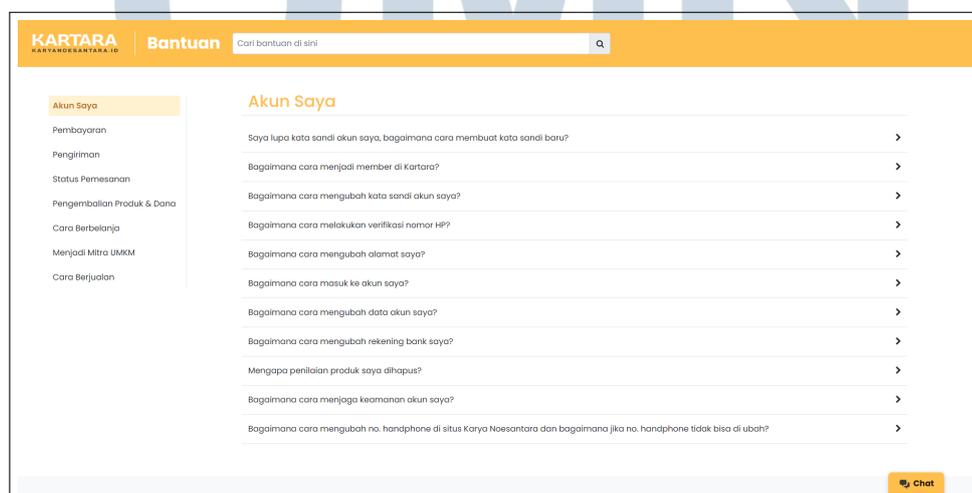
G. Halaman Bantuan

Dalam implementasinya, pengguna dapat melihat halaman bantuan yang berisikan kategori-kategori bantuan yang dapat diberikan oleh karyanoesantara.id. Tampilan awal halaman Bantuan dapat dilihat pada Gambar 3.31.



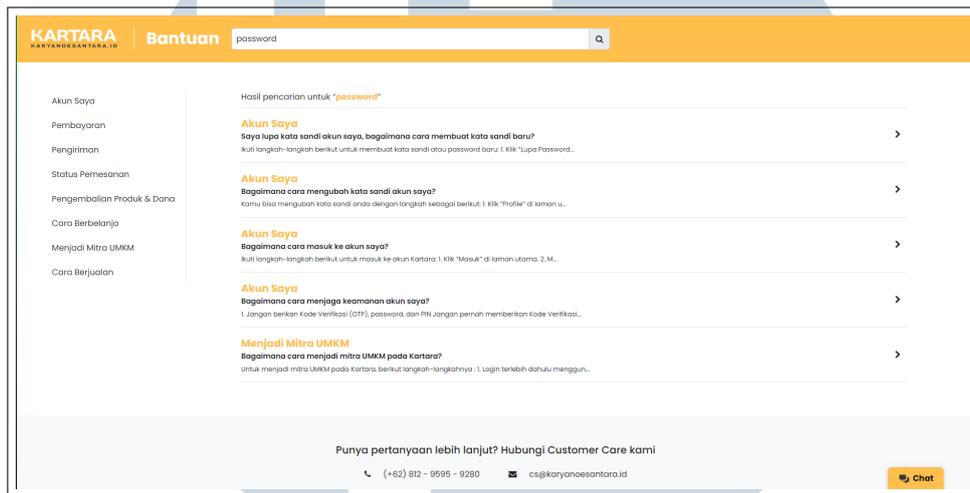
Gambar 3.31. Halaman Bantuan Awal

Dari Gambar 3.31, pengguna dapat melihat kategori bantuan apa saja yang tersedia. Jika pengguna mengklik salah satu kategori misalnya Akun Saya, pengguna akan diperlihatkan bantuan yang merupakan kategori dari bantuan akun pengguna. Dapat dilihat pada Gambar 3.32 untuk kategori bantuan Akun Saya.



Gambar 3.32. Halaman Bantuan Kategori Akun Saya

Terlihat dari Gambar 3.32, berbagai bantuan yang ditawarkan untuk kategori bantuan Akun Saya. Apabila pengguna kesulitan dalam mencari berdasarkan kategori bantuan, pengguna dapat mencari langsung bantuan yang diinginkan. Dengan menggunakan kolom pencarian, pengguna akan ditampilkan bantuan berdasarkan pencarian yang dilakukan seperti pada Gambar .



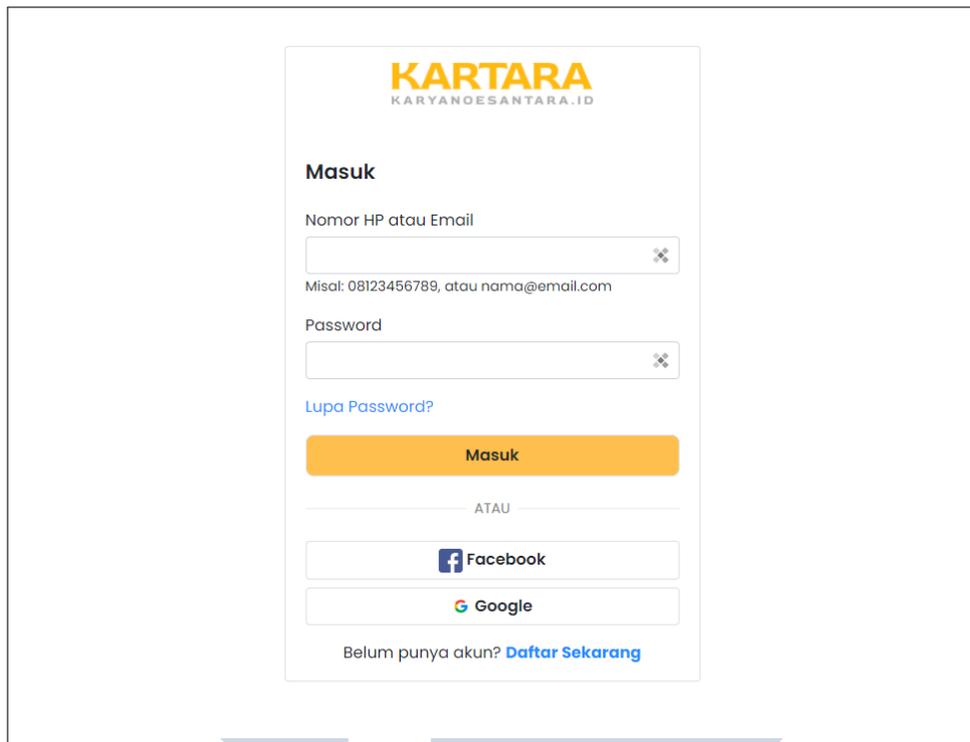
Gambar 3.33. Halaman Bantuan Pencarian untuk Kata Kunci "password"

Dari Gambar 3.33, pengguna dapat melihat semua bantuan yang ada yang berkaitan dengan kata kunci "password".

H. Lupa Password dan Ubah Password

Dalam implementasinya, ketika pengguna lupa *password* dari akun yang dimiliki, pengguna akan diminta kode verifikasi OTP-nya sesuai dengan yang digunakan pada akun-nya baik menggunakan *email* maupun nomor telepon. Pengguna yang telah berhasil di-verifikasi kode OTP-nya, akan masuk ke dalam akun-nya dan dapat mengubah *password*-nya pada halaman Profil.

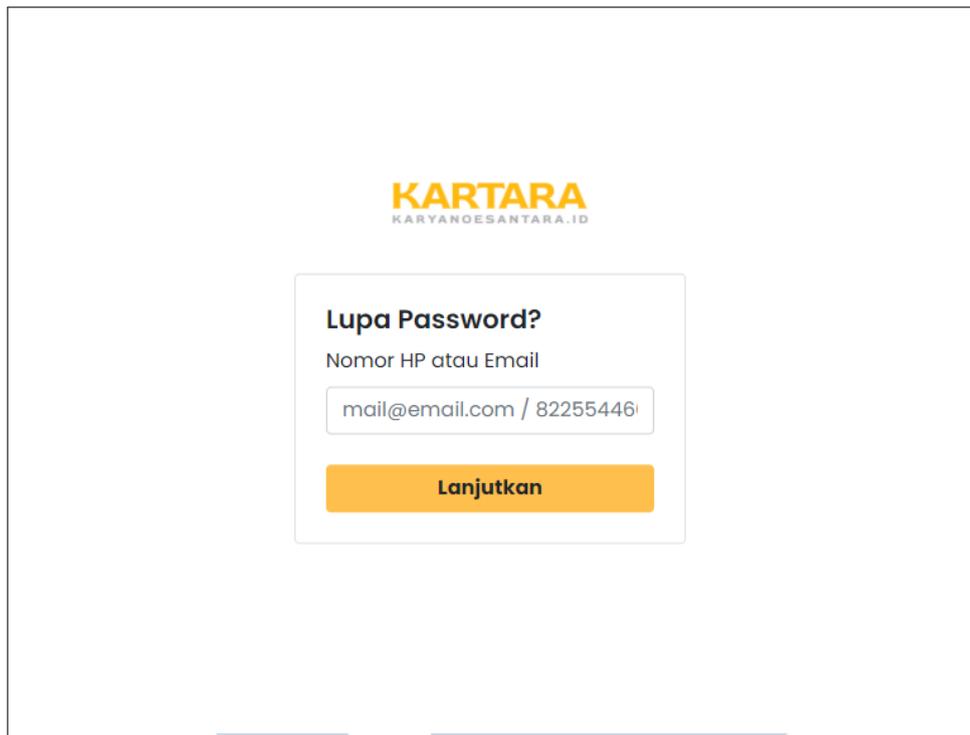
Berikut merupakan tampilan ketika pengguna ingin mengklik "Lupa Password?" pada halaman *Login* yang dapat dilihat pada Gambar 3.34.



Gambar 3.34. Halaman *Login*

Dari Gambar 3.34, ketika pengguna mengeklik tombol "Lupa Password?" pengguna akan diarahkan untuk memasukkan *email* atau nomor telepon sebagai alternatif dalam melakukan *login*. Tampilan ini dapat dilihat pada Gambar 3.35.

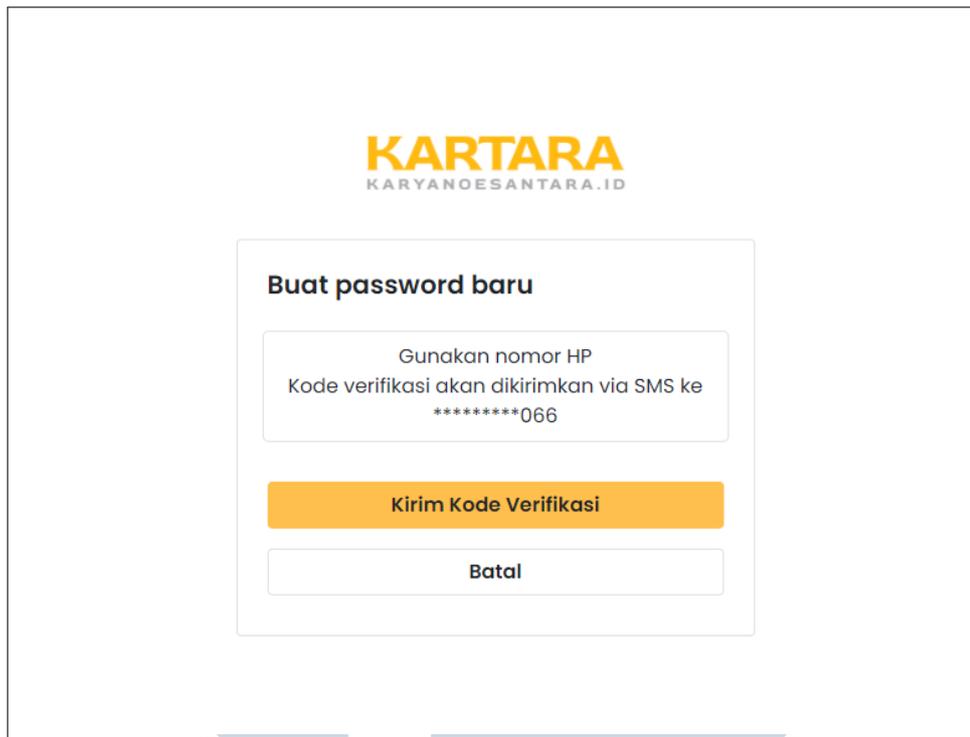
UMMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.35. Halaman Lupa Password Memasukkan *Email* atau Nomor Telepon

Dari Gambar 3.35, pengguna dapat memasukkan *email* atau nomor teleponnya. Apabila pengguna memasukkan nomor teleponnya maka sistem akan melakukan pengecekan terlebih dahulu apakah *email* atau nomor telepon yang dimasukkan ada di dalam *database*. Jika ada, maka pengguna akan ditanyakan terlebih dahulu untuk melakukan verifikasi. Tampilan ini dapat dilihat pada Gambar 3.36.

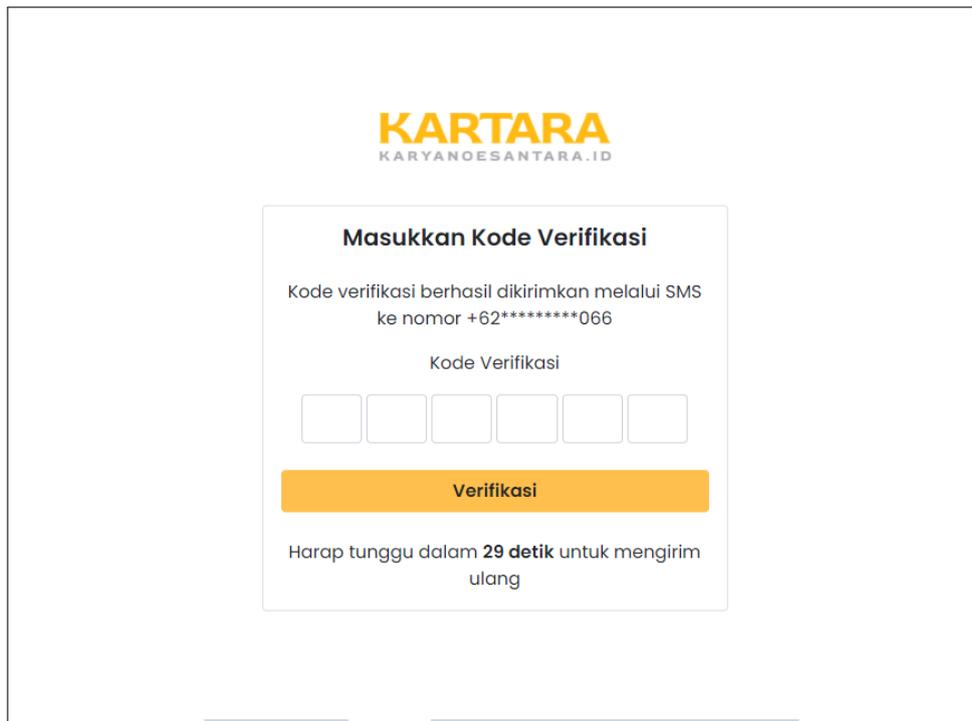
UIN
UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.36. Halaman Lupa Password Konfirmasi Verifikasi

Dari Gambar 3.36, ketika pengguna mengeklik tombol "Kirim Kode Verifikasi", pengguna akan diarahkan untuk memasukkan kode verifikasi OTP yang telah dikirimkan ke nomor telepon yang telah dimasukkan oleh pengguna sebelumnya. Pengiriman kode verifikasi OTP ini memanfaatkan Firebase dalam implementasinya. Tampilannya dapat dilihat pada Gambar 3.37

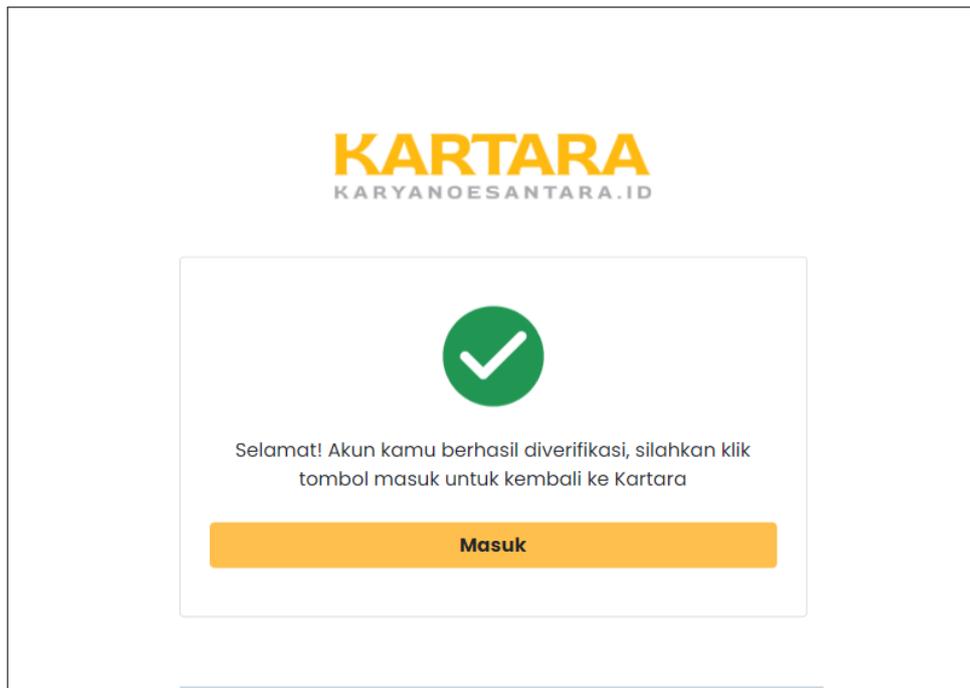
UIMIN
UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.37. Halaman Lupa Password Verifikasi Kode OTP

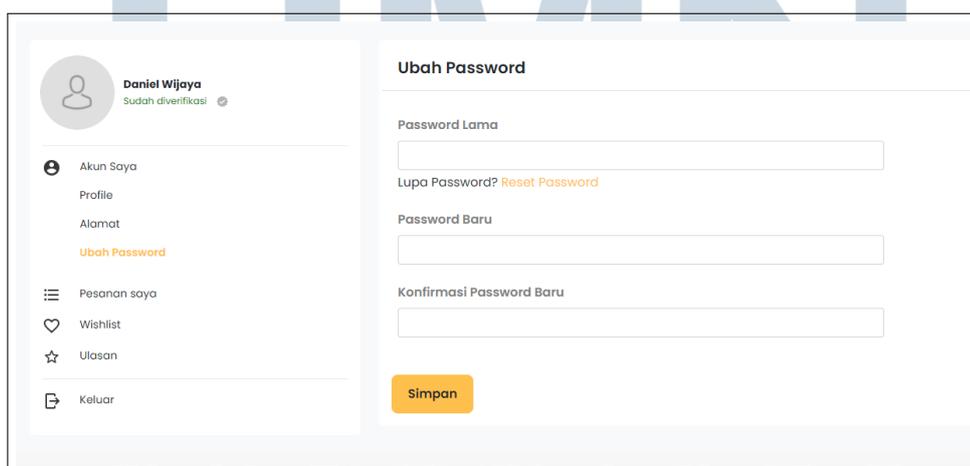
Dari Gambar 3.37, pengguna akan diminta verifikasi kode OTP yang telah dikirimkan ke nomor telepon pengguna. Setelah pengguna berhasil memasukkan kode verifikasi OTP, pengguna akan ditampilkan tampilan bahwa pengguna berhasil verifikasi. Dapat dilihat pada Gambar 3.38.

UMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.38. Halaman Lupa Password Berhasil Verifikasi

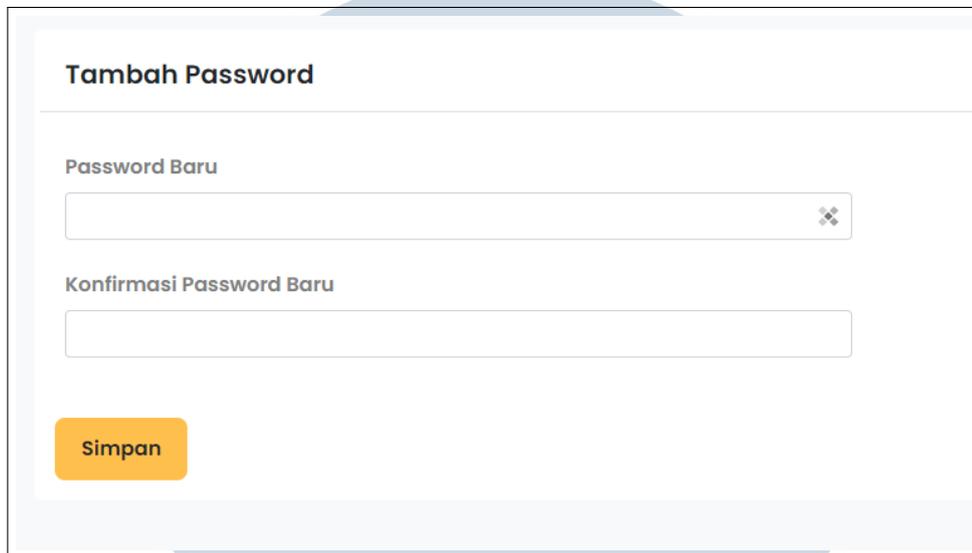
Dari Gambar 3.38, dapat dilihat bahwa pengguna berhasil diverifikasi. Pengguna kemudian akan diarahkan menuju halaman Profil, hal ini menandakan bahwa pengguna telah berhasil masuk ke dalam akun-nya. Untuk mengubah *password*, pengguna dapat membuka bagian Ubah Password dan mengubah *password*-nya secara langsung. Dapat dilihat pada Gambar 3.39.



Gambar 3.39. Halaman Profil Bagian Ubah Password

Pada Gambar 3.39, apabila pengguna lupa dengan *password*-nya, pengguna dapat melakukan *reset password* secara langsung. Pengguna akan diberikan

tampilan untuk langsung menambahkan *password*-nya dengan *password* yang baru. Dapat dilihat pada Gambar 3.40.



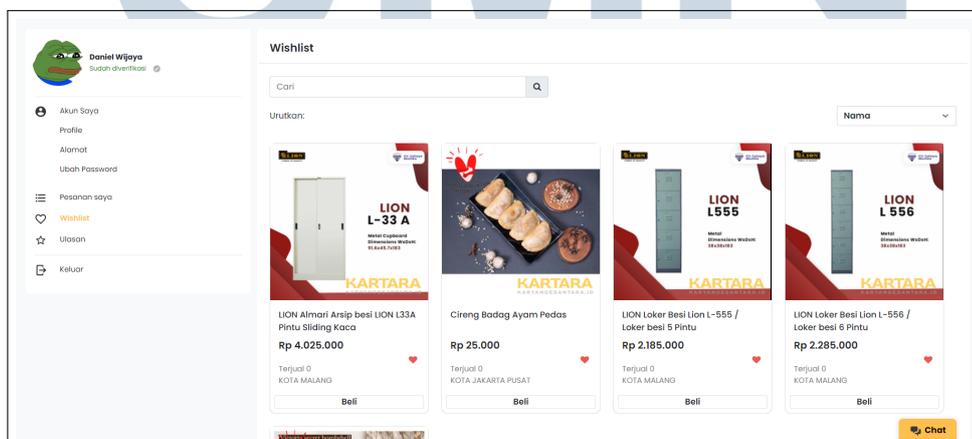
The screenshot shows a web form titled "Tambah Password". It contains two text input fields: "Password Baru" and "Konfirmasi Password Baru". Below the second field is an orange button labeled "Simpan".

Gambar 3.40. Halaman Profil Bagian Tambah Password

Pada Gambar 3.40, pengguna dapat langsung menambahkan *password* untuk akun-nya.

I. Fitur Wishlist

Pengguna dapat memasukkan *wishlist* untuk setiap produk cukup dengan mengklik tombol hati pada produk yang diinginkan. Pengguna dapat melihat daftar produk yang telah di-*wishlist* pada halaman Profil bagian Wishlist. Tampilan daftar *wishlist* pengguna dapat dilihat pada Gambar 3.41.

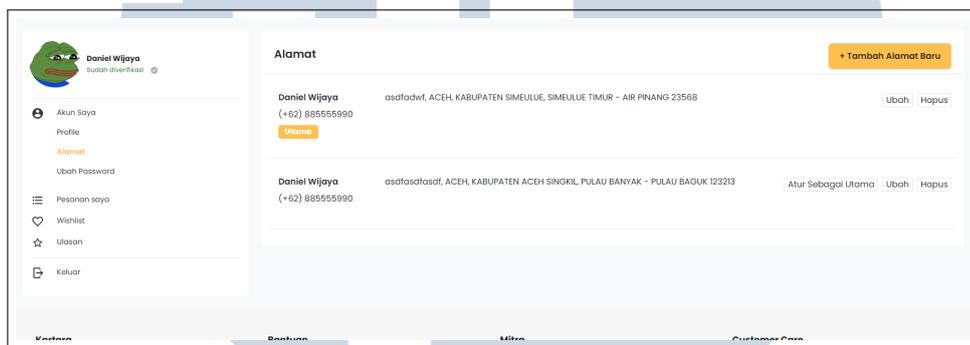


Gambar 3.41. Halaman Profil Bagian Wishlist

Pada Gambar 3.41, pengguna dapat melihat daftar produk yang telah di-*wishlist* termasuk informasi mengenai produk tersebut.

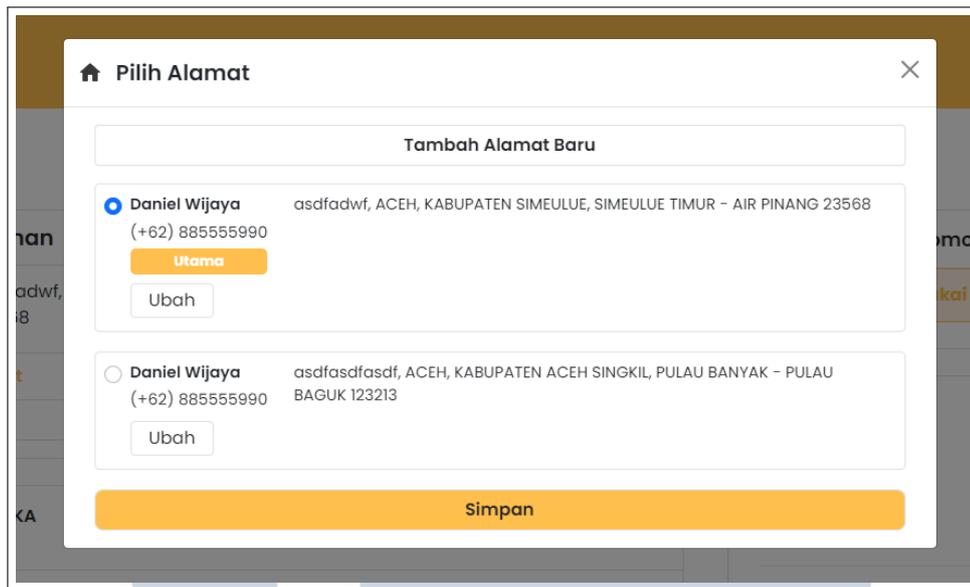
J. Manajemen Alamat Pengguna

Pengguna dapat mengelola alamat pengiriman-nya dengan dua cara. Cara pertama adalah mengelola alamat pengiriman di halaman Profil bagian Alamat. Pengguna dapat menambahkan, mengubah, menghapus, dan memilih alamat utama pada halaman ini. Tampilan dari halaman ini dapat dilihat pada Gambar ??.



Gambar 3.42. Ubah Alamat pada Halaman Profil

Pada Gambar 3.42, pengguna dapat mengelola alamat pengiriman-nya di halaman Profil. Cara kedua adalah dengan mengelola alamat pengiriman ketika pengguna sedang melakukan *checkout*. Pada halaman ini pengguna dapat menambahkan, mengubah, dan memilih alamat pengiriman untuk melakukan *checkout*. Pengguna tidak dapat menghapus maupun memilih alamat pengiriman utama. Tampilan dari halaman ini dapat dilihat pada Gambar 3.43.

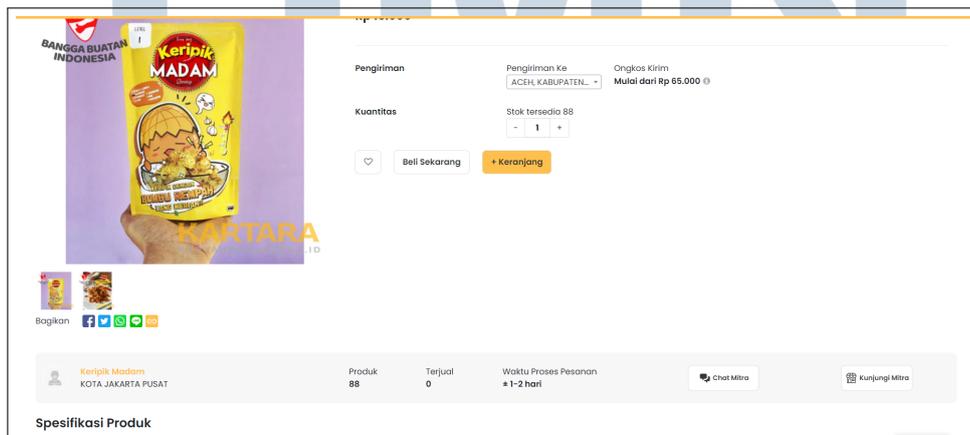


Gambar 3.43. Ubah Alamat pada Halaman *Checkout*

Pada Gambar 3.43, terlihat pengguna hanya dapat mengubah dan menambahkan alamat pengiriman-nya dan memilih alamat pengiriman untuk melakukan *checkout*.

K. Halaman Mitra

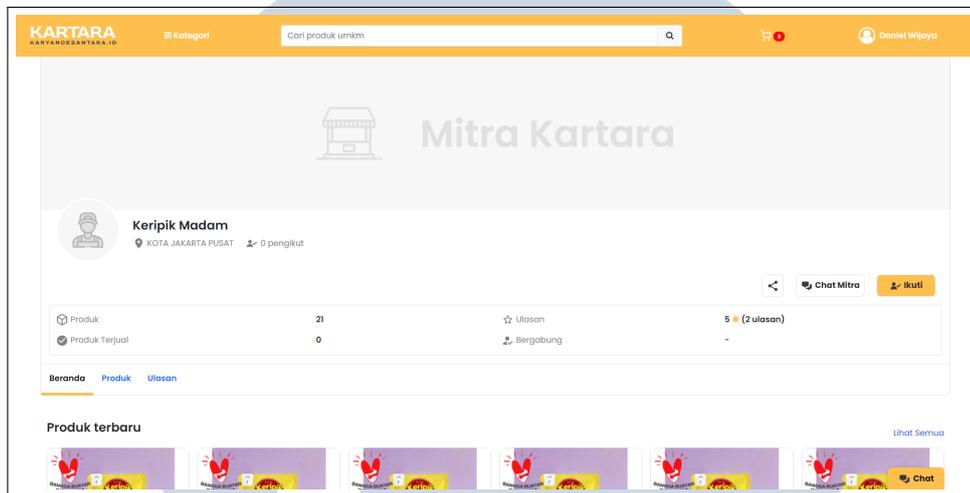
Pengguna dapat melihat informasi lengkap mengenai mitra tertentu. Untuk dapat melihat informasi mengenai mitra, pengguna dapat membuka halaman Detail Produk salah satu produk mitra tersebut, kemudian mengunjungi mitra tersebut, seperti yang dapat dilihat pada Gambar 3.44.



Gambar 3.44. Mengunjungi Mitra pada Halaman Detail Produk

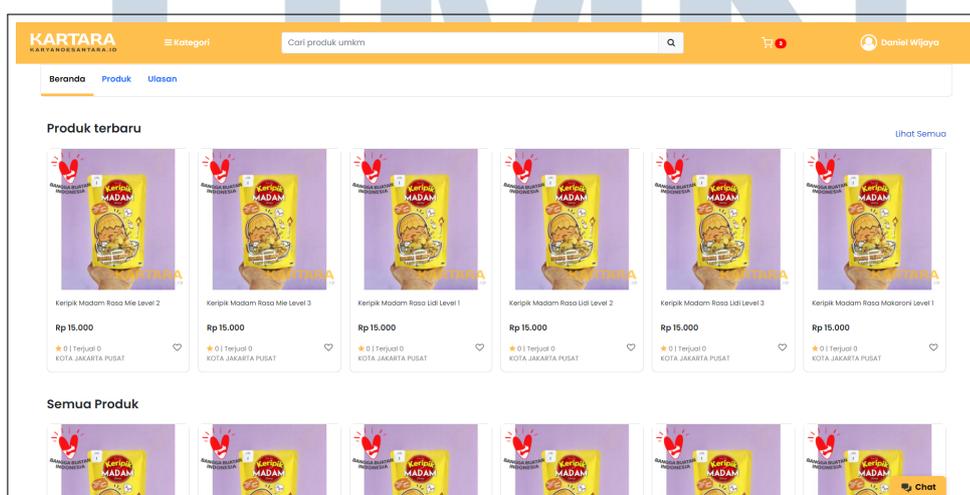
Dari Gambar 3.44, terlihat pengguna dapat mengunjungi mitra dengan

mengklik tombol "Kunjungi Mitra". Ketika pengguna mengklik tombol tersebut, pengguna akan diarahkan menuju halaman informasi mitra tersebut. Berikut merupakan tampilan halaman mitra yang dapat dilihat pada Gambar 3.45.



Gambar 3.45. Halaman Mitra

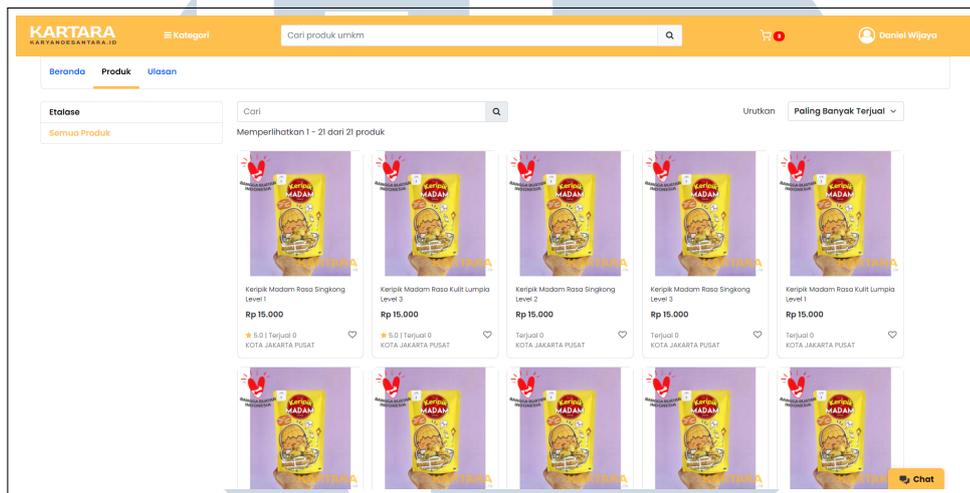
Dari Gambar 3.45, pengguna dapat melihat informasi dasar mengenai mitra tersebut. Pada bagian ini pengguna juga dapat mengikuti mitra, namun pada saat penulisan laporan ini, pengguna hanya mengikuti mitra tanpa menerima keuntungan apa pun dari mengikuti mitra seperti menerima notifikasi. Pada halaman mitra, terdapat 3 bagian halaman. Bagian pertama merupakan bagian Beranda yang dapat dilihat pada Gambar 3.46.



Gambar 3.46. Halaman Mitra Bagian Beranda

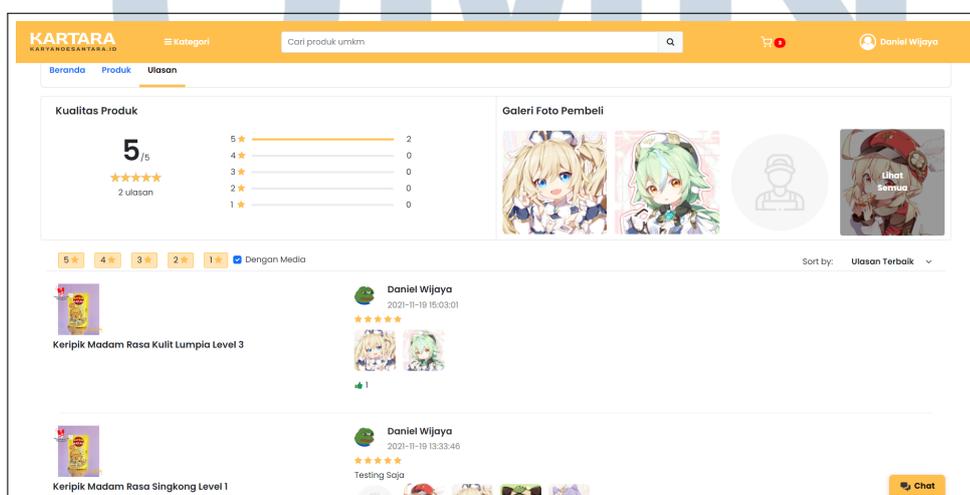
Dari Gambar 3.46, pengguna dapat melihat produk terbaru mitra dan produk

lainnya dari mitra tersebut. Bagian berikutnya adalah bagian Produk yang berisikan semua produk yang dijual oleh mitra tersebut. Untuk kategori produk yang dijual oleh mitra, untuk saat penulisan laporan ini masih hanya semua produk saja. Berikut merupakan tampilan dari bagian Produk pada halaman Mitra yang dapat dilihat pada Gambar 3.47.



Gambar 3.47. Halaman Mitra Bagian Produk

Dari Gambar 3.47, terlihat pengguna dapat melihat semua produk yang dijual oleh mitra. Pengguna juga dapat melakukan *sort* dan terdapat paginasi ketika produk ditampilkan. Untuk bagian ketika adalah bagian Ulasan. Pada bagian ini ditampilkan semua ulasan produk yang pernah diterima oleh mitra. Berikut merupakan tampilan pada bagian Ulasan pada halaman Mitra yang dapat dilihat pada Gambar 3.48.



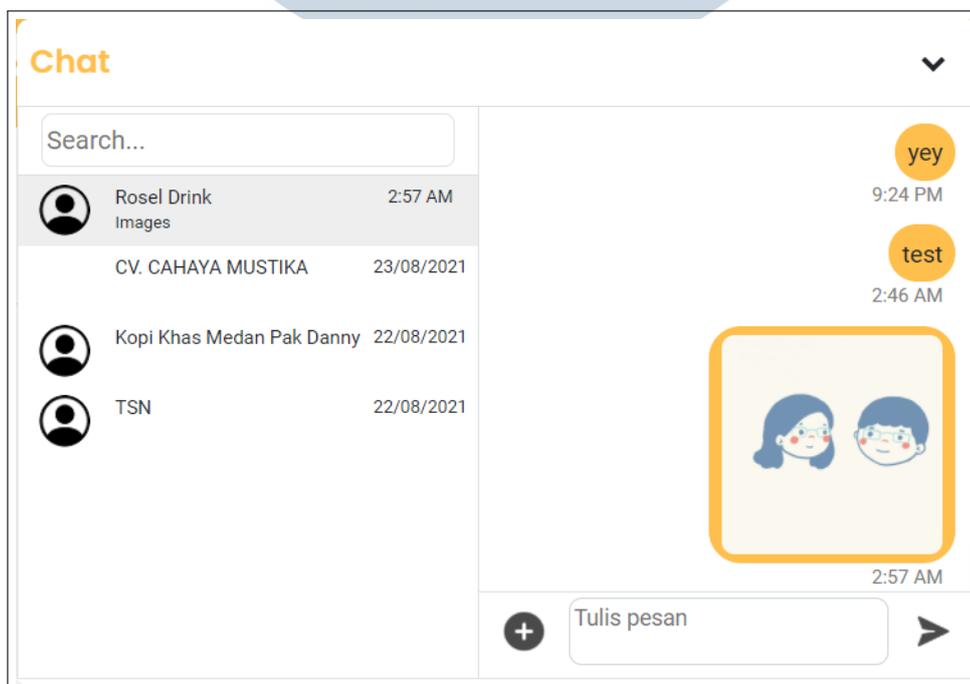
Gambar 3.48. Halaman Mitra Bagian Ulasan

Dari Gambar 3.48, pengguna dapat melihat rata-rata penilaian ulasan yang diterima oleh mitra. Pengguna juga dapat melihat galeri foto untuk setiap ulasan yang diterima oleh mitra tersebut. Pengguna juga dapat melihat semua ulasan yang pernah diterima oleh mitra. Dalam melihat ulasan ini, pengguna dapat menyembunyikan foto ulasan, mengurutkan ulasan, dan melihat ulasan dengan penilaian tertentu saja. Pengguna juga dapat menyukai ulasan dengan mengeklik ikon *like*.

L. Berkomunikasi Instan

Dalam implementasinya, digunakan *website* yang berbeda yang menggunakan Express.js sebagai *framework* utama dalam membangunnya. Agar pesan dapat dikirimkan dan diterima secara *real time*, dibutuhkan sistem *websocket* yang dapat menopang sistem komunikasi *down-time* yang sangat rendah atau tidak ada. Sistem *websocket* yang digunakan adalah *library* Socket.IO.

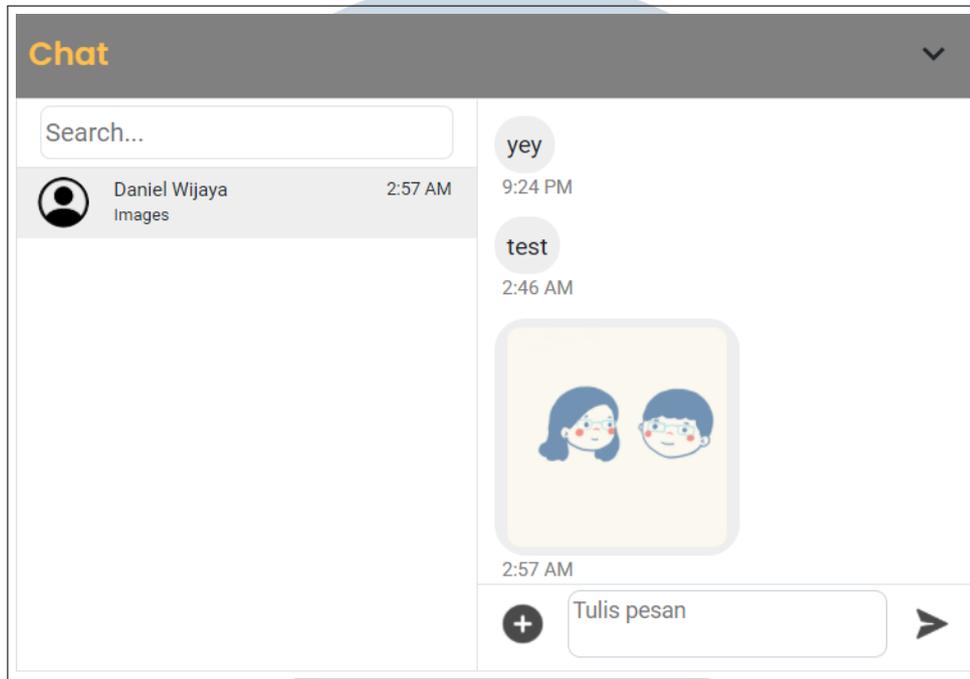
Berikut merupakan Gambar 3.49 ketika pengguna mengirimkan pesan teks dan gambar kepada mitra.



Gambar 3.49. Pengguna Mengirim Pesan Teks dan Gambar

Dari Gambar 3.49, dapat dilihat pengguna mengirimkan beberapa pesan teks dan juga gambar kepada mitra melalui *chat*. Mitra akan menerima pesan tersebut secara *real time*, maka ketika mitra sedang membuka atau tidak, mitra akan

menerima pesan tersebut. Pesan yang diterima oleh mitra dapat dilihat pada Gambar 3.50.



Gambar 3.50. Mitra Menerima Pesan Teks dan Gambar

Gambar 3.50 menunjukkan bahwa pesan berupa teks dan gambar telah berhasil diterima oleh mitra secara *real time*.

3.4 Kendala yang Ditemukan

Selama berlangsungnya magang, dihadapi beberapa kendala dalam implementasinya. Berikut penjabaran dari setiap kendala yang dihadapi.

3.4.1 Penggunaan Websockets untuk Real Time Message

Untuk mewujudkan sistem pengiriman pesan yang bersifat *real time*, dibutuhkan adanya penggunaan *websockets* untuk pengiriman dan penerimaan pesan. Namun, sistem *websockets* hanya dapat dijalankan pada *framework* yang menggunakan Node.js atau Composer sebagai *run-time environment* untuk menjalankan *web server*.

3.4.2 Fitur Filter Produk

Fitur *filter* produk yang telah dibuat masih menggunakan *Url* untuk melakukan *filter*. Hal ini menyebabkan fitur *filter* yang tidak bisa memilih lebih dari satu pilihan *filter*-nya. Penggunaan *filter* dengan cara ini juga membuat *website* menjadi kurang interaktif dikarenakan pengguna harus memuat ulang seluruh halaman *website* setiap saat *filter* dipilih.

3.4.3 Metode Pembayaran Pay Later

Selain berbagai metode pembayaran yang telah diimplementasikan, terdapat metode pembayaran yang tidak dapat diimplementasikan yaitu metode pembayaran *pay later*. Metode pembayaran ini tidak dapat diimplementasikan karena untuk menggunakan sistem *pay later*, hanya dapat menggunakan Xendit API. Xendit API memiliki fleksibilitas yang kurang dalam penggunaan sistem *pay later*-nya dikarenakan fiturnya yang masih baru.

3.5 Solusi atas Kendala yang Ditemukan

Dari kendala-kendala yang dihadapi, sebagian ditemukan solusi-nya dan sebagian belum ditemukan solusi-nya. Berikut pembabaran dari solusi untuk setiap kendala yang ditemukan.

3.5.1 Solusi Penggunaan Websockets untuk Real Time Message

Dikarenakan sistem *websockets* hanya dapat dijalankan pada framework yang menggunakan Node.js atau Composer, maka khusus untuk fitur mengirimkan pesan secara instan akan dibuat menggunakan *framework* Express.js. Untuk menampilkan aplikasi *chat* yang dibuat menggunakan *framework* Express.js, aplikasi *chat* akan ditampilkan dengan memanfaatkan elemen *iframe* dari HTML.

3.5.2 Solusi Fitur Filter Produk

Untuk membuat fitur *filter* produk yang lebih interaktif dan berfungsi sepenuhnya, fitur ini dibuat dengan memanfaatkan JQuery AJAX. Setiap saat pengguna mengubah *filter*, maka akan dilakukan panggilan data secara JQuery AJAX, kemudian data tersebut akan digunakan untuk menampilkan produk-produk sesuai *filter* yang dimasukkan pengguna.