

KateringConnect : Solusi Terintegrasi untuk Vendor Katering

Valdo Febrian¹, Anita Qoiriah²

^{1,2} Prodi S1-Teknik Informatika, Universitas Negeri Surabaya

¹valdo.20010@mhs.unesa.ac.id

²anitaqoiriah@unesa.ac.id

Abstrak— Perkembangan teknologi informasi yang terus berkembang menyebabkan terjadinya pergeseran pola hidup masyarakat secara masif. Salah satunya adalah kebiasaan dalam memesan makan secara daring. Industri makanan dan katering merupakan salah satu sektor yang terus berkembang, terutama di kota-kota besar. Kebutuhan akan layanan katering tidak hanya terbatas pada acara besar seperti pernikahan atau seminar, tetapi juga untuk keperluan sehari-hari seperti rapat bisnis, pertemuan keluarga, atau bahkan sekadar makan siang di kantor. Meskipun demikian, pencarian vendor katering terdekat dan sesuai preferensi masih susah dilakukan karena masyarakat masih perlu mencari satu per satu vendor katering di internet. Pada penelitian ini, dilakukan pembuatan aplikasi KateringConnect sebagai solusi terintegrasi untuk vendor katering sehingga memudahkan masyarakat dalam mencari vendor katering berdasarkan alamat *customer* atau kota yang dipilih. Aplikasi KateringConnect dikembangkan pada Android dengan metode pengembangan *Extreme Programming* (XP) yang dimulai dengan perencanaan, perancangan, *coding*, dan *testing*. *Testing* aplikasi dilakukan dengan metode *Black Box*. Hasil pengujian aplikasi dari sudut pandang *customer*, vendor, dan admin didapatkan hasil positif atau valid.

Kata Kunci— *Android*, *Aplikasi*, *Katering*, *Extreme Programming*, *Black Box*.

I. PENDAHULUAN

Di era digital yang semakin maju seperti saat ini, teknologi informasi telah memainkan peran yang sangat penting dalam berbagai aspek kehidupan manusia, termasuk dalam industri kuliner. Pada zaman yang serba cepat dan mobilitas yang tinggi, kebutuhan akan solusi praktis dalam pemesanan makanan sangat diperlukan. Masyarakat modern menginginkan kemudahan dalam mengakses layanan kuliner yang cepat, fleksibel, dan dapat diandalkan, terutama bagi mereka yang memiliki kesibukan tinggi dan waktu terbatas untuk menyiapkan makanan sendiri.

Pemesanan makanan secara daring atau online telah menjadi tren yang terus meningkat dalam beberapa tahun terakhir[1]. Masyarakat kini cenderung lebih memilih untuk memesan makanan melalui aplikasi mobile atau situs web daripada secara konvensional[2]. Hal ini disebabkan oleh faktor-faktor seperti kesibukan masyarakat yang meningkat, perubahan gaya hidup, dan kebutuhan akan kemudahan serta efisiensi dalam berbagai aspek kehidupan. Sebelum era digital, proses pemesanan makanan biasanya dilakukan dengan menelepon restoran, mengunjungi restoran, atau menggunakan layanan pesan-antar makanan yang belum terintegrasi secara digital. Namun, dengan meningkatnya penggunaan internet dan ponsel pintar, konsumen kini dapat dengan mudah mengakses berbagai

pilihan makanan dari berbagai restoran tanpa harus keluar rumah. Aplikasi dan platform pemesanan makanan online menawarkan kenyamanan yang belum pernah ada sebelumnya, memungkinkan pengguna untuk melihat menu secara langsung, membaca ulasan, membandingkan harga, dan bahkan memantau status pengiriman secara real-time.

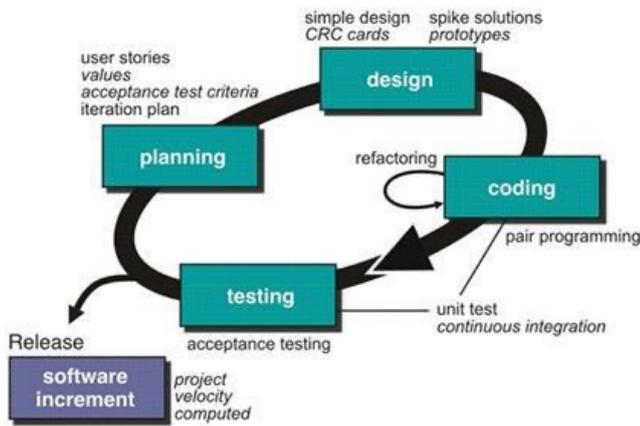
Industri makanan dan katering merupakan salah satu sektor yang terus berkembang, terutama di kota-kota besar. Kebutuhan akan layanan katering tidak hanya terbatas pada acara besar seperti pernikahan atau seminar, tetapi juga untuk keperluan sehari-hari seperti rapat bisnis, pertemuan keluarga, atau bahkan sekadar makan siang di kantor. Acara-acara tersebut sering kali melibatkan banyak tamu, di mana penyediaan makanan yang memadai dan bervariasi sangat penting. Layanan katering memungkinkan penyelenggara acara untuk mengalihkan fokus dari persiapan makanan ke aspek lain dalam perencanaan acara, dengan mengandalkan para profesional untuk menangani segala hal mulai dari menu hingga penyajian. Selain itu, layanan katering juga memberikan fleksibilitas untuk menyesuaikan menu sesuai dengan kebutuhan khusus, seperti preferensi diet, alergi makanan, atau tema acara, sehingga dapat memenuhi berbagai permintaan khusus dari klien.

Namun, masyarakat yang tinggal di kota-kota besar atau daerah tertentu kerap kali mengalami kesulitan dalam mencari katering terdekat dan sesuai preferensi mereka. Masyarakat di kota-kota besar sering dihadapkan dengan terlalu banyak pilihan katering yang ditampilkan di berbagai platform, namun pilihan tersebut belum tentu relevan dengan kebutuhan mereka. Orang mungkin kesulitan menemukan layanan katering yang benar-benar sesuai dengan anggaran, selera, atau kebutuhan khusus mereka seperti katering halal, preferensi diet, alergi makanan, atau tema acara. Hal ini dapat memakan banyak waktu karena mereka perlu mencari satu-satu di internet. Sedangkan di daerah pedesaan atau kota kecil, tantangan lain muncul karena terbatasnya jumlah katering yang tersedia. Akses untuk menemukan katering di daerah ini juga sulit, karena sebagian besar layanan tidak terdaftar di platform pencarian besar atau aplikasi pesan antar makanan. Akibatnya, orang sering mengandalkan rekomendasi dari orang-orang terdekat atau mencari secara manual tanpa informasi pendukung yang memadai. Dan juga saat ini, baik di kota-kota besar atau daerah masih banyak vendor-vendor katering yang belum memasarkan katering mereka secara daring atau online, dari katering yang dijalankan oleh individu atau bisnis yang sudah besar maupun yang masih berkembang. Pemasaran katering pun terbatas dan tidak menjangkau pengguna secara luas, oleh karenanya keberadaan katering mereka kurang

diketahui oleh masyarakat. Oleh karena itu, diperlukan adanya suatu aplikasi pemesanan catering yang dapat mawadahi vendor-vendor catering dan membantu customer mendapatkan rekomendasi berdasarkan kota yang dipilih atau sesuai alamatnya.

II. METODE PENELITIAN

A. Metode Pengembangan Aplikasi



Gbr. 1 Metode Extreme Programming

Dalam penelitian ini, metode perancangan aplikasi yang digunakan adalah *Extreme Programming* (XP). Metode XP adalah pendekatan fleksibel untuk pengembangan perangkat lunak dan manajemen proyek yang bertujuan untuk menghasilkan Software yang lebih baik[3]. Metode XP sangat peka terhadap perubahan keinginan pengguna karena siklus pengembangannya yang pendek. Secara garis besar metode ini seperti pada Gbr.1 mempunyai langkah-langkah sebagai berikut :

- 1) *Planning*: Pada tahap *planning*, didefinisikan kebutuhan-kebutuhan yang diperlukan untuk merancang dan membangun aplikasi berbasis *android* unruk mawadahi proses pemesanan catering antara vendor dan *customer*.
- 2) *Design*: Tahap ini merupakan tahap pembuatan desain sebuah sistem. Desain yang digunakan dalam penelitian ini menggunakan model UML yaitu Use Case Diagram dan Sequence Diagram, desain *database* menggunakan ERD (Entity Relationship Diagram), dan desain antarmuka *low fidelity*. Desain yang dibuat merupakan representasi dari sistem yang sedang dikembangkan.
- 3) *Coding*: Tahap *coding* disini merupakan tahapan pengembangan aplikasi *android* dari sebuah desain aplikasi menjadi kode program. Penulisan kode program pada aplikasi ini menggunakan Android Studio yang merupakan IDE resmi untuk pengembangan *android* dan bahasa pemrograman Kotlin.
- 4) *Testing*: Sebelum sistem digunakan oleh pengguna, pengujian sistem dilakukan untuk mencari kesalahan. Proses ini sangat penting untuk mengetahui seberapa

baik sistem dirancang. Jika ada peluang untuk memunculkan dan menemukan kesalahan yang belum diketahui, pengujian dianggap baik dan berhasil. Bukan untuk menghilangkan kesalahan, tetapi untuk mengidentifikasi jumlah kesalahan yang mungkin ada dalam sistem. Dalam penelitian ini menggunakan metode pengujian Black Box. Metode ini digunakan untuk menemukan kesalahan dan menunjukkan fungsionalitas aplikasi saat dioperasikan, menunjukkan apakah input diterima dengan benar dan output yang dihasilkan sesuai dengan yang diharapkan.

B. Planning/Perencanaan

- 1) *Kebutuhan Data*: Kebutuhan data mengacu pada spesifikasi yang mendefinisikan informasi dan data yang diperlukan untuk mendukung fungsi dan operasi suatu sistem. Berikut data yang dibutuhkan dalam aplikasi ini:
 - a. Data *customer* yang terdiri dari nama, kota, alamat, nomor telepon, foto KTP, dan foto selfie KTP.
 - b. Data vendor catering yang terdiri dari nama catering, kota, alamat, nomor telepon, foto KTP, foto selfie KTP, foto dapur, dan menu-menanya.
- 2) *Kebutuhan Fungsional dan Non-Fungsional*: Kebutuhan fungsional mengacu pada spesifikasi apa yang harus dilakukan oleh sistem untuk memenuhi tujuan bisnis dan kebutuhan pengguna[4]. Ini mencakup interaksi antara pengguna dan sistem serta proses yang harus dilakukan untuk mencapai hasil tertentu. Sedangkan kebutuhan non-fungsional adalah kebutuhan yang berfokus pada properti perilaku pada sistem dan juga sebagai batasan layanan atau fungsi yang ditawarkan oleh sistem[5]. Kebutuhan Fungsional dan Non-Fungsional bagi *customer*, vendor, dan admin dapat dilihat pada Tabel I, Tabel II, dan Tabel III.

TABEL I
KEBUTUHAN FUNGSIONAL DAN NON-FUNGSIONAL CUSTOMER

| Fungsional | Non-Fungsional |
|----------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1. Melakukan registrasi akun. | 1.1 Memasukkan email, password, dan data diri. 1.2 Tidak dapat menggunakan email yang sudah didaftarkan sebelumnya. 1.3 Menggunakan email dengan format email seperti @gmail.com. 1.4 Membuat kata sandi minimal 8 karakter. |
| 2. Mengunggah foto KTP dan foto Selfie | 2.1 Foto menggunakan format .jpg |

| Fungsional | Non-Fungsional |
|--------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| KTP saat registrasi akun. | |
| 3. Melakukan Login | 3.1 Memasukkan email dan password yang telah terdaftar |
| 4. Melihat vendor-vendor catering sesuai kota yang dipilih atau alamat mereka. | 4.1 Pada fitur pencarian, Customer hanya dapat mencari berdasarkan nama vendor atau kategori menu yang ingin dicari |
| 5. Melihat data dan menu dari vendor catering. | 5.1 Tidak dapat memilih menu yang dinonaktifkan oleh vendor. |
| 6. Melakukan kustomisasi menu. | 6.1 Memilih setidaknya satu opsi dari masing-masing kustom menu. 6.2. Mengisi jumlah porsi pesanan diatas minimal pemesanan yang ditentukan oleh vendor. |
| 7. Melakukan pemesanan dan pembayaran catering. | 7.1 Memilih jadwal pengantaran minimal 1 hari sebelum acara atau kegiatan. 7.2 Memilih metode pembayaran tunai atau KC Wallet. |
| 8. Penambahan porsi. | 8.1 Hanya dapat menambah porsi pada pesanan yang berstatus proses. |
| 9. Melakukan Top Up dan tarik dana KC Wallet. | 9.1 Saat <i>top up</i> perlu memasukkan nominal dengan minimal yang ditentukan oleh admin. 9.2 Saat tarik dana perlu memasukkan nominal dengan minimal yang ditentukan oleh admin dan saldo mencukupi. |
| 10. Konfirmasi pesanan telah diterima. | 10.1 Customer perlu mengonfirmasi bahwa pesanan telah diterima dengan batas waktu 1 hari setelah vendor mengantarkan pesanan. |
| 11. Melihat riwayat pesanan dan memberikan penilaian | 11.1 Hanya dapat memberikan penilaian pada pesanan yang berstatus selesai. 11.2 Hanya dapat memberikan penilaian dengan opsi satu hingga lima bintang, tidak bisa setengah bintang atau lebih dari lima bintang. |

TABEL II
KEBUTUHAN FUNGSIONAL DAN NON-FUNGSIONAL VENDOR

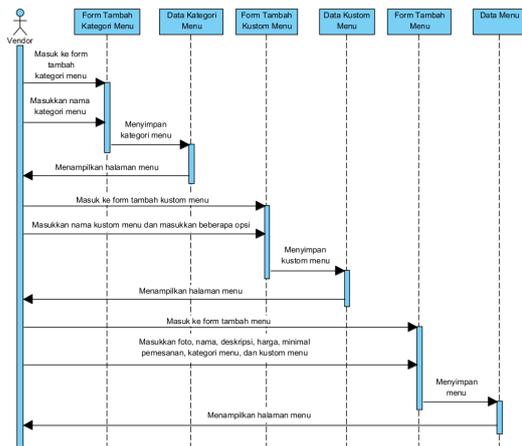
| Fungsional | Non-Fungsional |
|------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1. Melakukan registrasi akun. | 1.1 Memasukkan email, password, dan data diri. 1.2 Tidak dapat menggunakan email yang sudah didaftarkan sebelumnya. 1.3 Menggunakan email dengan format email seperti @gmail.com. 1.4 Membuat kata sandi minimal 8 karakter. |
| 2. Mengunggah foto KTP dan foto Selfie KTP saat registrasi akun. | 2.1 Foto menggunakan format .jpg |
| 3. Melakukan Login | 3.1 Memasukkan email dan password yang telah terdaftar. |
| 4. Melihat, menambah, mengubah, menghapus kategori menu. | 4.1 Ketika menghapus kategori menu, akan menghapus seluruh menu yang terdapat dalam kategori tersebut. |
| 5. Melihat, menambah, mengubah, menghapus kustom menu. | |
| 6. Melihat, menambah, mengubah, menghapus menu. | |
| 7. Melihat dan memproses pesanan masuk. | 7.1 Harus melakukan konfirmasi pesanan diterima atau ditolak kepada <i>customer</i> dengan batas waktu 1 hari. |
| 8. Menyelesaikan pesanan. | 8.1. Harus mengirim foto bukti pengiriman. |
| 9. Melihat riwayat pesanan. | |
| 10. Melakukan Top Up dan tarik dana KC Wallet. | 10.1 Saat <i>top up</i> perlu memasukkan nominal dengan minimal yang ditentukan oleh admin. 10.2 Saat tarik dana perlu memasukkan nominal dengan minimal yang ditentukan oleh admin dan saldo mencukupi. |

TABEL III
KEBUTUHAN FUNGSIONAL DAN NON-FUNGSIONAL ADMIN

| Fungsional | Non-Fungsional |
|---------------------------------------------|-------------------------------------------------------------------------|
| 1. Melakukan Login | 1.1 Harus menggunakan email dan password khusus admin. |
| 2. Melihat dan memverifikasi data pengguna. | 2.1 Harus melakukan konfirmasi pendaftaran diterima atau ditolak kepada |

ketersediaannya secara otomatis oleh sistem, sedangkan data diri pengguna akan diverifikasi secara manual oleh admin. Setelah verifikasi berhasil, *database* akan menyimpan data tersebut. Selanjutnya dapat memasukkan *email* dan *password* melalui form *login*, yang selanjutnya dikirim ke *database* untuk diverifikasi. Setelah verifikasi berhasil, halaman akan diarahkan menuju halaman beranda.

b. *Sequence Diagram Menambahkan Menu*



Gbr. 4 Sequence Diagram Menambahkan Menu

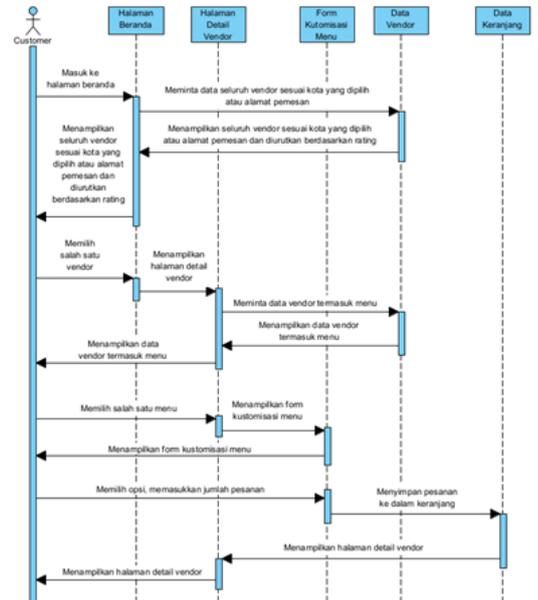
Pada Gbr. 4 merupakan *sequence diagram* untuk proses menambahkan menu, vendor mengirimkan data menu ke *database* melalui *form* tambah menu. Data menu yang perlu diisi ialah foto, nama, deskripsi, harga, minimal pemesanan, kategori, dan kustom menu. Kategori bersifat wajib untuk ditambahkan ke dalam menu. Jika belum terdapat kategori, vendor dapat menambahkannya terlebih dahulu dengan menyimpan nama kategori ke *database* melalui *form* tambah kategori menu.

Untuk kustom menu bersifat opsional untuk ditambahkan ke dalam menu. Kustom menu dapat dilakukan jika vendor ingin membuat sebuah kustom menu terkait spesifikasi atau bahan pada menu tertentu yang dapat dipilih oleh *customer* dengan tambahan biaya maupun tanpa tambahan biaya.

c. *Sequence Diagram Simpan Pesanan ke Keranjang*

Setelah *login*, *customer* akan diarahkan ke halaman beranda. Halaman beranda menampilkan seluruh vendor sesuai kota yang dipilih atau alamat *customer* dan diurutkan berdasarkan *rating*. Selanjutnya

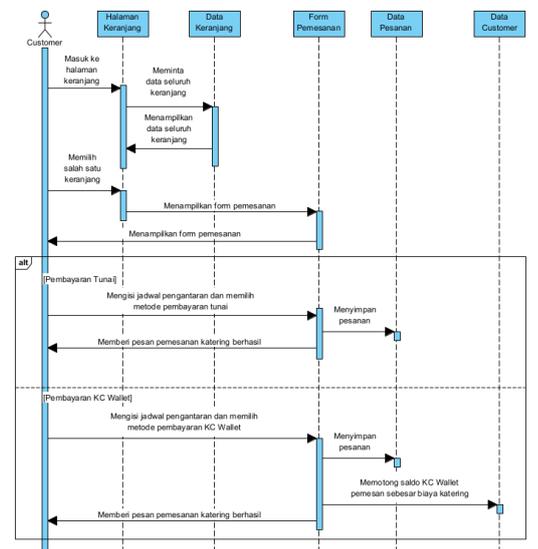
customer perlu memilih salah satu vendor untuk dapat melihat menu-menu yang ditawarkan oleh vendor. Untuk menyimpan pesanan ke dalam keranjang, *customer* perlu memilih menu yang diinginkan dan melakukan kustomisasi menu jika ada dan memasukkan jumlah pesanan. Setelah semua terpenuhi, pesanan akan disimpan ke dalam



database keranjang. *Sequence Diagram* simpan pesanan ke keranjang dapat dilihat pada Gbr. 5.

Gbr. 5 Sequence Diagram Simpan Pesanan ke Keranjang

d. *Sequence Diagram Checkout Keranjang*

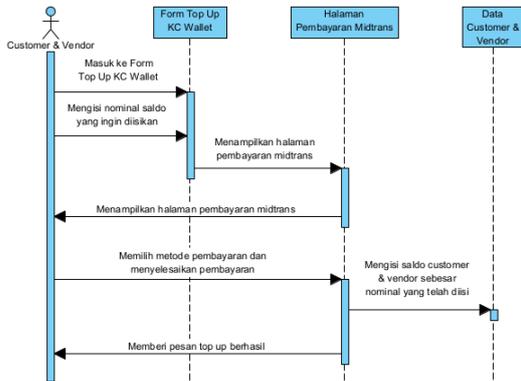


Gbr. 6 Sequence Diagram Checkout Keranjang

Pada Gbr. 6 merupakan *sequence diagram* untuk proses *checkout* keranjang, setelah menyimpan pesanan ke dalam

keranjang, customer dapat melakukan pemesanan catering dengan cara *checkout* keranjang miliknya. *Customer* dapat masuk ke halaman keranjang dan memilih keranjang yang ingin di-*checkout*. Kemudian, customer perlu mengisi jadwal pengantaran dan memilih metode pembayaran. Setelah terpenuhi, pesanan akan disimpan ke dalam *database* pesanan.

e. *Sequence Diagram Top Up KC Wallet*



Gbr. 7 Sequence Diagram Top Up KC Wallet

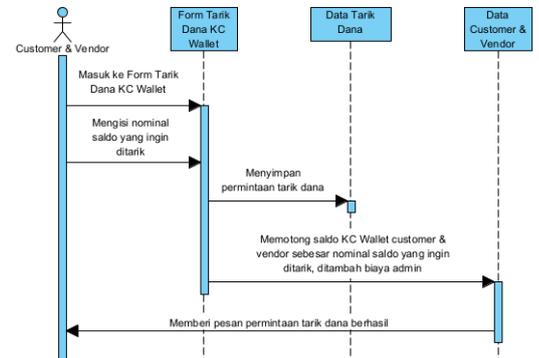
Pada Gbr. 7 merupakan *sequence diagram* untuk proses *top up* KC Wallet. Pada aplikasi ini disediakan dompet digital bernama KC Wallet, yang berguna untuk memudahkan *customer* ketika ingin melakukan pembayaran dengan non-tunai. Untuk mengisi KC Wallet, pengguna dapat masuk ke *form top up* KC Wallet dan mengisi nominal saldo. Selanjutnya pengguna akan diarahkan ke halaman pembayaran Midtrans. Pada halaman tersebut, pengguna dapat memilih metode pembayaran dan menyelesaikan pembayarannya. Ketika pembayaran telah berhasil, saldo akan diisikan ke KC Wallet pengguna dan memberi pesan kepada pengguna bahwa *top up* telah berhasil.

Tidak hanya untuk customer, *top up* KC Wallet juga diperlukan bagi vendor ketika dompet mereka minus akibat menyelesaikan pesanan dengan pembayaran tunai. Karena setiap pesanan akan dikenakan biaya layanan kepada vendor. Ketika KC Wallet vendor minus atau kurang dari nol, akibatnya adalah vendor tidak akan ditampilkan ke *customer* dan tidak dapat dicari, dengan kata lain vendor tidak akan dapat menerima pesanan.

f. *Sequence Diagram Tarik Dana KC Wallet*

Untuk menarik dana KC Wallet, pengguna dapat masuk ke *form* tarik dana KC Wallet dan mengisi nominal saldo.

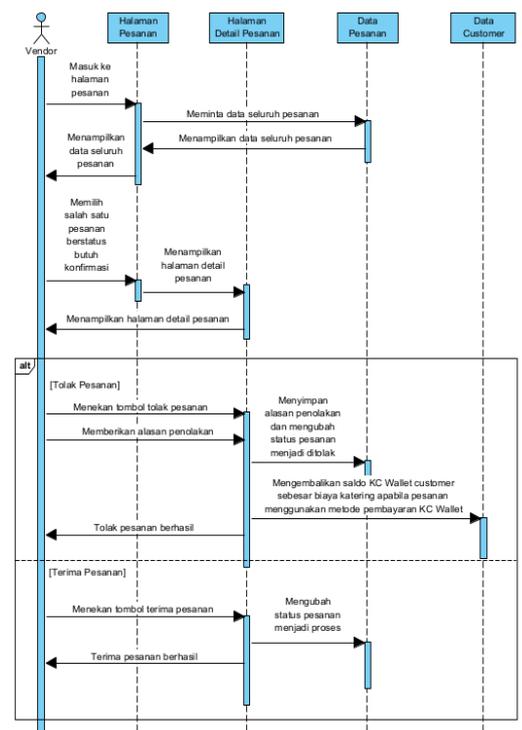
Kemudian permintaan tarik dana akan disimpan dan akan disampaikan ke admin. Selanjutnya saldo KC Wallet pengguna akan dipotong sebesar nominal saldo yang ingin ditarik, ditambah biaya admin dan mendapatkan pesan bahwa permintaan tarik



dana telah berhasil. *Sequence Diagram* tarik dana KC Wallet dapat dilihat pada Gbr. 8.

Gbr. 8 Sequence Diagram Tarik Dana KC Wallet

g. *Sequence Diagram Menerima Pesanan*



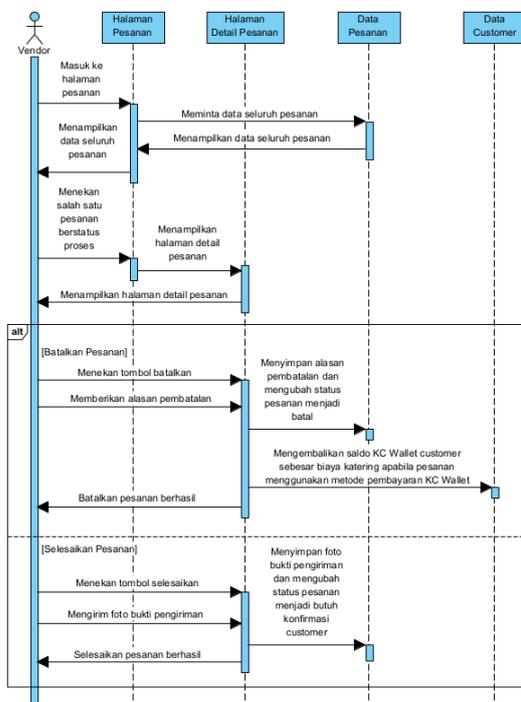
Gbr. 9 Sequence Diagram Menerima Pesanan

Pada Gbr. 9 merupakan *sequence diagram* untuk proses menerima pesanan. Ketika terdapat pesanan masuk, vendor perlu melakukan konfirmasi dengan menerima atau menolak pesanan. Hal tersebut dapat dilakukan dengan cara masuk ke halaman

pesanan, selanjutnya akan ditampilkan data seluruh pesanan. Kemudian memilih salah satu pesanan berstatus butuh konfirmasi, lalu halaman detail pesanan akan ditampilkan dan terdapat tombol untuk terima pesanan dan tolak pesanan.

Ketika tombol terima pesanan ditekan, status pesanan tersebut akan diubah menjadi proses. Dan ketika tombol tolak pesanan ditekan, vendor perlu memberikan alasan penolakan, selanjutnya alasan penolakan akan disimpan dan status pesanan akan diubah menjadi ditolak. Kemudian, saldo KC Wallet *customer* akan dikembalikan sebesar biaya catering apabila *customer* menggunakan metode pembayaran KC Wallet.

h. *Sequence Diagram Menyelesaikan Pesanan oleh Vendor*



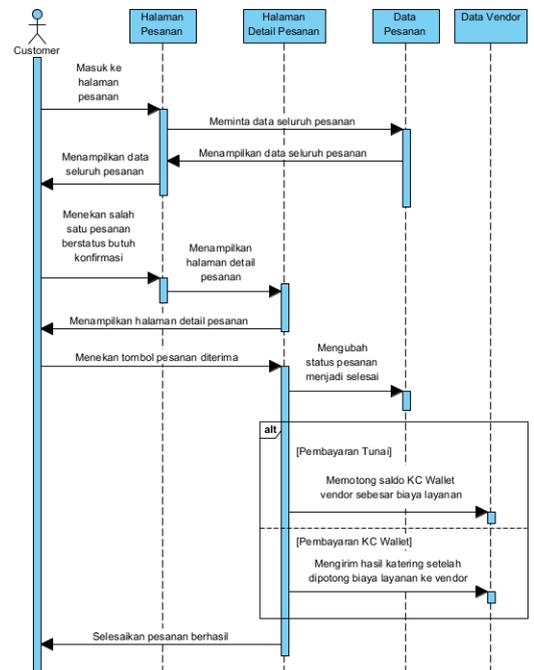
Gbr. 10 Sequence Diagram Menyelesaikan Pesanan oleh Vendor

Pada Gbr. 10 merupakan *sequence diagram* untuk proses menyelesaikan pesanan oleh vendor. Pesanan dapat diselesaikan oleh vendor dengan cara membatalkan atau menyelesaikan pesanan. Hal tersebut dapat dilakukan dengan cara masuk ke halaman pesanan, selanjutnya akan ditampilkan data seluruh pesanan. Kemudian menekan salah satu pesanan berstatus proses, lalu halaman detail pesanan akan ditampilkan

dan terdapat tombol untuk selesaikan dan batalkan pesanan.

Ketika tombol selesaikan pesanan ditekan, vendor perlu mengirim foto bukti pengiriman, selanjutnya foto bukti pengiriman akan disimpan dan status pesanan akan diubah menjadi butuh konfirmasi pemesan. Dan ketika tombol batalkan pesanan ditekan, vendor perlu memberikan alasan pembatalan, selanjutnya alasan pembatalan akan disimpan dan status pesanan akan diubah menjadi batal. Kemudian, saldo KC Wallet *customer* akan dikembalikan sebesar biaya catering apabila *customer* menggunakan metode pembayaran KC Wallet.

i. *Sequence Diagram Menyelesaikan Pesanan oleh Customer*

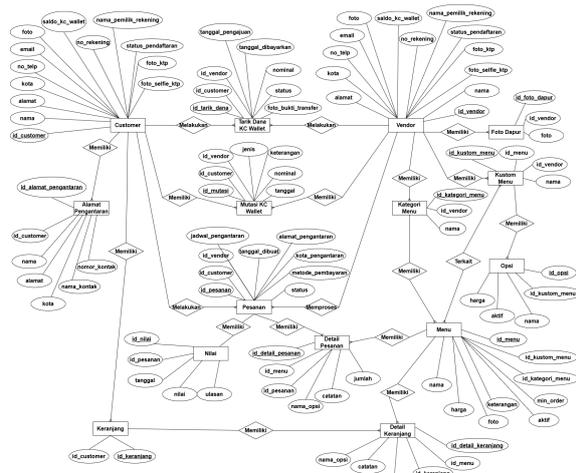


Gbr. 11 Sequence Diagram Menyelesaikan Pesanan oleh Customer

Pada Gbr. 11 merupakan *sequence diagram* untuk proses menyelesaikan pesanan oleh *customer*. Ketika pesanan telah diselesaikan oleh vendor, *customer* perlu memberikan konfirmasi bahwa pesanan telah diterima dengan batas waktu 1 hari. Apabila telah melewati batas waktu, pesanan akan otomatis diselesaikan. Hal tersebut dapat dilakukan dengan cara masuk ke halaman pesanan, selanjutnya akan ditampilkan data seluruh pesanan. Kemudian menekan salah satu pesanan berstatus butuh konfirmasi, lalu halaman detail pesanan akan ditampilkan dan terdapat tombol pesanan diterima.

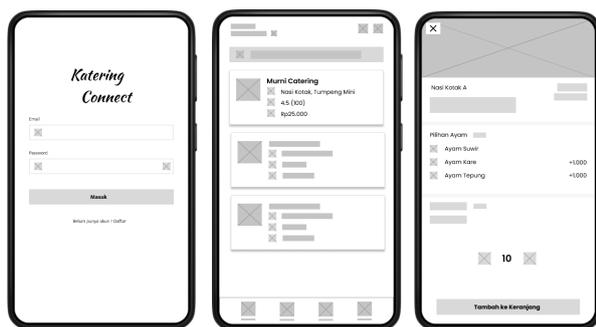
Ketika tombol pesanan diterima ditekan, status pesanan akan diubah menjadi selesai. Selanjutnya, hasil biaya catering setelah dipotong biaya layanan akan dikirimkan ke vendor jika pesanan tersebut menggunakan pembayaran KC Wallet. Namun jika pesanan tersebut menggunakan pembayaran tunai, saldo KC Wallet vendor akan dipotong sebesar biaya layanan.

- 3) *Entity Relationship Diagram: Entity Relationship Diagram (ERD)* berguna untuk menggambarkan bagaimana berbagai komponen dalam sistem berinteraksi dan berhubungan satu sama lain[8]. *Entity Relationship Diagram* dari aplikasi ini dapat dilihat pada Gbr. 12.



Gbr. 12 Entity Relationship Diagram

- 4) *Rancangan Antarmuka: Rancangan antarmuka menggunakan Low Fidelity Prototype*, representasi dasar dan sederhana dari suatu produk atau desain[9]. Rancangan antarmuka dari aplikasi ini dapat dilihat pada Gbr. 13.



Gbr. 13 Rancangan Antarmuka

D. *Testing/Pengujian*

Sebelum sistem digunakan oleh pengguna, pengujian sistem dilakukan untuk mencari kesalahan. Proses ini sangat penting untuk mengetahui seberapa baik sistem dirancang. Jika ada

peluang untuk memunculkan dan menemukan kesalahan yang belum diketahui, pengujian dianggap baik dan berhasil. Bukan untuk menghilangkan kesalahan, tetapi untuk mengidentifikasi jumlah kesalahan yang mungkin ada dalam sistem. Dalam penelitian ini menggunakan metode pengujian *Black Box*. Metode ini digunakan untuk menemukan kesalahan dan menunjukkan fungsionalitas aplikasi saat dioperasikan, menunjukkan apakah input diterima dengan benar dan output yang dihasilkan sesuai dengan yang diharapkan[10].

III. HASIL DAN PEMBAHASAN

Bab ini berisi tentang hasil yang didapatkan dalam penelitian dengan topik “KateringConnect: Solusi Terintegrasi untuk Vendor Katering” disertai juga dengan pembahasannya. Pembahasan pada bab ini adalah proses bisnis dari aplikasi KateringConnect. Berikut pembahasan pada bab ini:

A. *Pemesanan Katering*

Untuk pemesanan catering terdapat rangkaian proses yang dilakukan oleh *customer*, *vendor*, dan *admin*. Masing-masing dari proses tersebut dijelaskan sebagai berikut:

- 1) *Vendor Registrasi: Tahapan Register* perlu dilakukan jika vendor belum memiliki akun, di sini vendor diminta untuk mengisi data diri serta mendaftarkan akun mereka. Beberapa data yang perlu dimasukkan oleh vendor yaitu nama, kota, alamat, nomor telepon, *email*, dan *password*, serta memilih jenis akun yaitu akun vendor. Setelah semua telah terpenuhi, vendor dapat menekan tombol daftar. Selanjutnya Firebase akan memverifikasi data secara otomatis dan apabila verifikasi berhasil, Firebase akan mendaftarkan akun baru dan menyimpan data pada *Database*, selanjutnya data vendor seperti foto KTP dan foto selfie KTP akan diverifikasi secara manual oleh admin. Hasil tampilan dari halaman *register* dapat dilihat pada Gbr. 14.



Gbr. 14 Halaman Register

- 2) *Admin Verifikasi Data Vendor*: Data pendaftaran yang telah dikirimkan oleh vendor akan diterima oleh admin untuk diverifikasi keasliannya seperti foto KTP dan foto *selfie* KTP.



Gbr. 15 Halaman Verifikasi Data

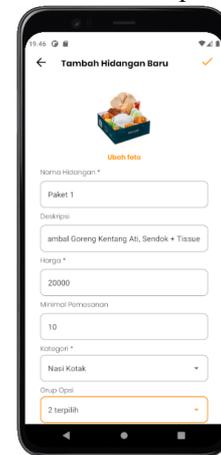
Gbr. 15 merupakan hasil tampilan dari halaman verifikasi data yang menampilkan data pengguna baru yang terdiri dari nama, *email*, kota, alamat, nomor telepon, foto KTP, dan foto *selfie* KTP dan juga terdapat tombol untuk menerima dan menolak pendaftaran. Ketika admin menerima pendaftaran, maka pengguna sudah dapat menggunakan akunnya. Ketika admin menolak pendaftaran, maka pengguna perlu memperbaiki data yang ditolak oleh admin, selanjutnya perbaikan tersebut akan diverifikasi oleh admin.

- 3) *Vendor Login*: Tahapan *Login* perlu dilakukan jika vendor telah memiliki akun, di sini vendor perlu memasukkan *email* dan *password* yang telah didaftarkan untuk dapat melanjutkan ke halaman utama vendor. Setelah terpenuhi, vendor dapat menekan tombol masuk. Selanjutnya, Firebase akan memverifikasi *email* dan *password* yang vendor masukkan. Ketika verifikasi berhasil, selanjutnya akan berpindah ke halaman utama sesuai jenis akun tersebut, jika tersimpan sebagai akun vendor maka akan dikirim ke halaman utama vendor, dan jika tersimpan sebagai akun *customer* maka akan dikirim ke halaman utama *customer*. Hasil tampilan dari halaman login dapat dilihat pada Gbr. 16.



Gbr. 16 Halaman Login

- 4) *Vendor Menambahkan Menu*: Pada tahapan ini, vendor menambahkan menu yang nantinya dapat dilihat dan dipesan oleh *customer*. Pada halaman tambah menu terdapat beberapa data menu yang perlu ditambahkan yaitu foto, nama, deskripsi, harga, minimal pemesanan, kategori, dan grup opsi (kustom menu). Tetapi untuk deskripsi, minimal pemesanan, dan grup opsi bersifat opsional, boleh diisi boleh tidak. Untuk minimal pemesanan jika tidak diisi akan otomatis bernilai satu. Setelah semua telah terpenuhi, vendor dapat menekan tombol centang. Selanjutnya foto akan disimpan pada *Firestore Storage* dan menyimpan URInya dan data-data menu pada *Database* *Firestore*. Hasil tampilan dari halaman



tambah menu dapat dilihat pada Gbr. 17.

Gbr. 17 Halaman Tambah Menu

- 5) *Customer Registrasi*: Tahapan *Register* perlu dilakukan jika *customer* belum memiliki akun, di sini *customer* diminta untuk mengisi data diri serta mendaftarkan akun mereka. Dan untuk tampilan dari halaman *Register* dapat dilihat pada Gbr. 14. Bagi *customer*, pastikan pada pemilihan jenis akun memilih akun pembeli, dan mengisi seluruh data diri. Setelah

semua terpenuhi, pengguna dapat menekan tombol daftar. Kemudian halaman akan berpindah ke halaman utama *customer*.

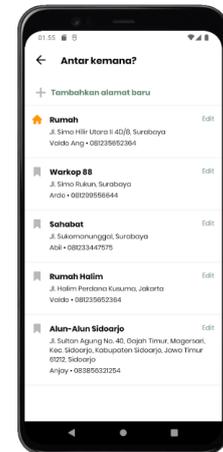
- 6) *Admin Verifikasi Data Customer*: Data pendaftaran yang telah dikirimkan oleh *customer* akan diterima oleh admin untuk diverifikasi keasliannya seperti foto KTP dan foto *selfie* KTP. Ketika admin menerima pendaftaran, maka pengguna sudah dapat menggunakan akunnya. Ketika admin menolak pendaftaran, maka pengguna perlu memperbaiki data yang ditolak oleh admin, selanjutnya perbaikan tersebut akan diverifikasi oleh admin. Untuk tampilan dari halaman verifikasi data dapat dilihat pada Gbr. 15.
- 7) *Customer Login*: Tahapan *Login* perlu dilakukan jika *customer* telah memiliki akun, di sini *customer* perlu memasukkan *email* dan *password* yang telah didaftarkan untuk dapat melanjutkan ke halaman utama *customer*. Dan untuk tampilan dari halaman *Login* dapat dilihat pada Gbr. 16.
- 8) *Customer Mencari Vendor Sesuai Alamat*: Ketika telah *Login*, halaman yang akan *customer* lihat pertama kali adalah halaman beranda. Pada halaman beranda menampilkan vendor-vendor terdekat sesuai kota *customer*. Gbr. 18 merupakan hasil tampilan dari halaman beranda yang menampilkan foto, nama, kategori menu, penilaian, dan ongkos kirim dari masing-masing vendor. Dan juga terdapat kotak pencarian untuk mencari vendor berdasarkan nama vendor atau kategori menu dan saldo KC Wallet.



Gbr. 18 Halaman Beranda Customer

- 9) *Customer Mencari Vendor Luar Kota*: Selain menampilkan vendor yang sesuai dengan kota *customer*, aplikasi ini juga dapat menampilkan vendor pada kota atau alamat lain dengan menekan tombol *Antar Ke* pada bagian atas kiri halaman beranda yang dapat dilihat pada Gbr. 18, kemudian halaman alamat pengantaran yang dapat dilihat pada Gbr. 19 akan ditampilkan. Alamat pengantaran yang baru akan digunakan ketika *customer* menekan salah satu alamat

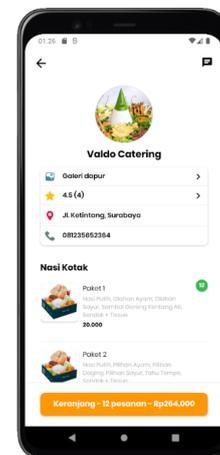
yang telah disimpan, dan halaman akan kembali ke halaman beranda dan menampilkan seluruh vendor sesuai alamat yang telah dipilih. Dan juga ongkos kirim dari setiap vendor akan diperbarui sesuai jarak antara vendor dan alamat *customer* yang telah dipilih. Secara default, alamat pengantaran menggunakan



alamat utama yang ditambahkan saat registrasi akun.

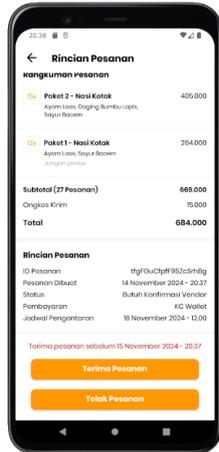
Gbr. 19 Halaman Alamat Pengantaran

- 10) *Customer Memilih Menu*: Setelah memilih salah satu vendor, halaman rincian vendor yang dapat dilihat pada Gbr. 20 akan terbuka yang menampilkan data-data vendor dan menu-menu yang dapat dipesan. Beberapa data vendor yang ditampilkan ialah foto, nama, alamat, nomor telepon vendor. Kemudian terdapat tombol galeri dapur dan Rating vendor yang mengarah ke masing-masing halamannya. Dan juga tombol Keranjang untuk checkout keranjang atau memesan catering.



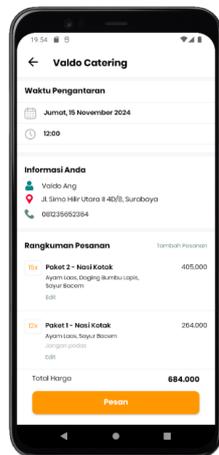
Gbr. 20 Halaman Rincian Vendor

11) *Customer Kustomisasi Menu*: Setelah memilih salah satu menu, halaman kustomisasi menu yang dapat dilihat pada Gbr. 21 akan terbuka yang menampilkan data menu, grup opsi (kustom menu) yang perlu diperlu dipilih salah satu oleh *customer* (jika ada), mengisi catatan untuk vendor yang bersifat opsional, dan menentukan jumlah porsi. Beberapa data menu yang ditampilkan ialah foto, nama, deskripsi, dan harga menu. Kemudian terdapat tombol Tambah ke Keranjang yang ketika ditekan akan menyimpan pesanan ke dalam keranjang.



Gbr. 21 Halaman Kustomisasi Menu

12) *Customer Memesan Katering*: Ketika ingin memesan katering, *customer* perlu melakukan *checkout* keranjang dengan menekan tombol Keranjang yang dapat dilihat pada Gbr.20.

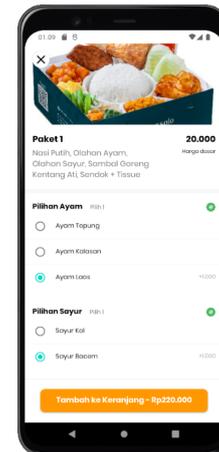


Gbr. 22 Halaman Pemesanan

Kemudian halaman pemesanan yang dapat dilihat pada Gbr. 22 akan terbuka yang menampilkan informasi *customer*, rangkuman pesanan, total harga yang perlu dibayar, dan metode pembayaran. Beberapa informasi *customer* yang ditampilkan ialah nama, alamat, dan nomor telepon, lalu terdapat rangkuman pesanan yang berisi jumlah porsi, nama menu, nama opsi yang dipilih, catatan, *subtotal*,

ongkos kirim dan total harga. Pada halaman ini, *customer* juga diminta untuk mengisi waktu pengantaran dan memilih metode pembayaran. Setelah terpenuhi, *customer* baru dapat memesan katering tersebut.

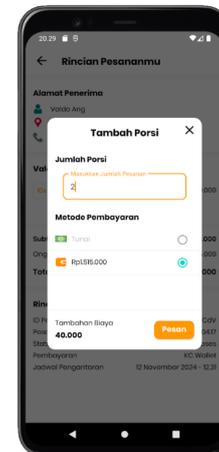
13) *Vendor Konfirmasi Pesanan Masuk*: Setelah mendapatkan pesanan masuk, vendor perlu memberikan konfirmasi untuk menerima atau menolak pesanan dengan cara menekan tombol terima atau tolak pesanan pada Gbr. 23. Vendor diberi batas waktu 1 hari untuk memberikan konfirmasi setelah *customer* memesan katering.



Gbr. 23 Halaman Konfirmasi Pesanan Masuk

Gbr. 23 merupakan hasil tampilan dari halaman konfirmasi pesanan masuk yang menampilkan menu yang dipesan, informasi pembeli, rincian pesanan, dan terdapat tombol untuk menerima dan menolak pesanan.

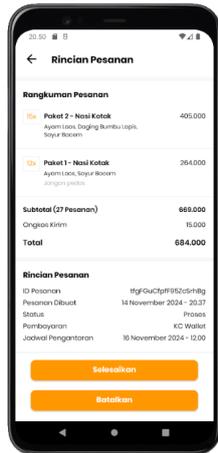
14) *Customer Menambah Porsi (Opsional)* : Pada aplikasi ini, *customer* dapat menambah porsi ketika pesanan masih berstatus proses, tetapi tidak dapat mengurangi jumlah porsi. Hal ini dapat membantu customer ketika terjadi suatu hal yang mengharuskan customer menambah jumlah porsi.



Gbr. 24 Dialog Tambah Jumlah Porsi

Gbr. 24 merupakan hasil tampilan dari *dialog* tambah jumlah porsi yang terdapat inputan jumlah porsi yang perlu diisi *customer* kemudian tambahan biaya ditampilkan. Setelah terpenuhi, *customer* dapat menekan tombol pesan.

- 15) *Vendor Menyelesaikan Pesanan*: Pada tahapan ini, vendor perlu menyiapkan dan mengantarkan pesanan sesuai permintaan dan waktu pengantarannya. Setelah selesai, vendor dapat menyelesaikan pesannya dalam aplikasi. Selain menyelesaikan pesanan, vendor juga dapat membatalkan pesanan apabila terdapat kondisi yang mengharuskannya dengan cara menekan tombol selesaikan atau batalkan pada Gbr. 25.



Gbr. 25 Halaman Selesaikan Pesanan

- 16) *Customer Konfirmasi Pesanan Diterima*: Setelah pesanan telah diantar dan diselesaikan oleh vendor, pemesan perlu mengonfirmasi bahwa pesanan telah diterima. *Customer* diberi batas waktu 1 hari untuk memberikan konfirmasi setelah pesanan diselesaikan oleh vendor. Apabila melebihi batas waktu, pesanan akan otomatis diterima *customer*.



Gbr. 26 Halaman Terima Pesanan

Gbr. 26 merupakan hasil tampilan dari halaman terima pesanan yang menampilkan alamat pengiriman, menu yang dipesan, rincian pesanan, bukti pengiriman,

batas waktu menerima pesanan, dan tombol untuk menerima pesanan.

B. *KC Wallet*

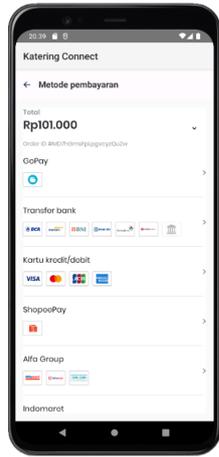
Pada aplikasi ini disediakan dompet digital bernama KC Wallet, yang berguna untuk memudahkan melakukan pembayaran untuk pemesanan catering. Terdapat beberapa proses dalam KC Wallet yaitu Top Up, tarik dana, dan transfer dana oleh admin. Berikut penjelasan dari masing-masing proses tersebut:

- 1) *Customer dan Vendor Top Up*: Pada aplikasi ini disediakan dompet digital bernama KC Wallet, yang berguna untuk memudahkan *customer* ketika ingin melakukan pembayaran dengan non-tunai. Sebelum memesan catering, tentu saja saldo KC Wallet *customer* perlu diisi dengan cara top up. Tidak hanya untuk *customer*, top up KC Wallet juga diperlukan bagi vendor ketika dompet mereka minus akibat menyelesaikan pesanan dengan pembayaran tunai. Karena setiap pesanan akan dikenakan biaya layanan kepada vendor. Ketika KC Wallet vendor minus atau kurang dari nol, akibatnya adalah vendor tidak akan ditampilkan ke *customer* dan tidak dapat dicari, dengan kata lain vendor tidak akan dapat menerima pesanan.



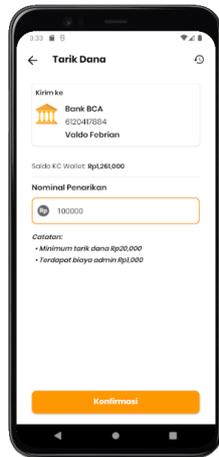
Gbr. 27 Halaman Top Up KC Wallet

Pada Gbr. 27 merupakan hasil tampilan dari halaman *top up* KC Wallet yang terdapat inputan *nominal top up* yang perlu diisi oleh pengguna dan tombol konfirmasi. Selain itu, juga terdapat catatan *minimum top up* dan biaya admin *top up*. Setelah inputan telah diisi dengan *minimum* yang telah ditentukan, pengguna dapat menekan tombol konfirmasi. Kemudian halaman pembayaran Midtrans yang dapat dilihat pada Gbr. 28 akan terbuka. Pengguna dapat memilih metode pembayaran yang diinginkan dan menyelesaikan pembayarannya. Setelah pembayaran berhasil, saldo akan diisi ke KC Wallet pengguna.



Gbr. 28 Halaman Pembayaran Midtrans

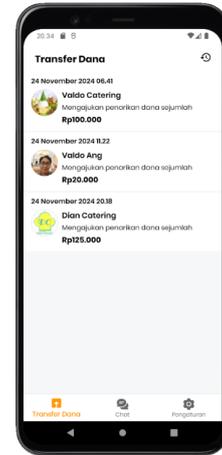
- 2) *Customer dan Vendor Tarik Dana*: Selain top up KC Wallet yang berfungsi untuk mengisi saldo KC Wallet, adapun tarik dana KC Wallet yang berfungsi untuk menarik saldo KC Wallet ke rekening bank pengguna.



Gbr. 29 Halaman Tarik Dana KC Wallet

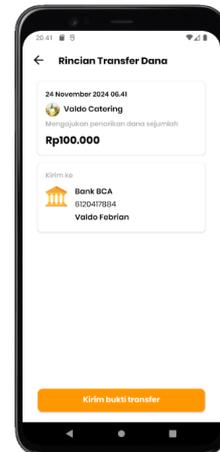
Pada Gbr. 29 merupakan hasil tampilan dari halaman tarik dana KC Wallet yang terdapat inputan nominal penarikan yang perlu diisi oleh pengguna dan tombol konfirmasi. Selain itu, juga terdapat informasi rekening tujuan penarikan, saldo KC Wallet saat ini, catatan *minimum* tarik dana dan biaya admin tarik dana. Setelah inputan telah diisi dengan *minimum* yang telah ditentukan dan saldo mencukupi, pengguna dapat menekan tombol konfirmasi. Kemudian saldo pengguna akan berkurang sebesar nominal penarikan ditambah biaya admin, dan permintaan penarikan dana akan dikirimkan ke admin.

- 3) *Admin Transfer Dana*: Permintaan tarik dana KC Wallet dari pengguna akan diterima oleh admin, dan admin bertugas untuk mentransfer dana ke rekening bank pengguna.



Gbr. 30 Halaman Transfer Dana

Pada Gbr. 30 merupakan hasil tampilan dari halaman transfer dana yang menampilkan ringkasan dari seluruh permintaan tarik dana oleh pengguna. Ringkasan dari masing-masing permintaan tarik dana yang ditampilkan ialah tanggal pengajuan, foto dan nama pengguna, dan nominal penarikan. Ketika salah satu permintaan tarik dana ditekan, akan menuju ke halaman rincian transfer dana yang dapat dilihat pada Gbr. 31.



Gbr. 31 Halaman Rincian Transfer Dana

Gbr. 31 merupakan hasil tampilan dari halaman rincian transfer dana yang menampilkan tanggal pengajuan, foto dan nama pengguna, nominal penarikan, rekening bank tujuan transfer, dan tombol kirim bukti transfer. Disini admin perlu mentransfer dana ke rekening bank pengguna sesuai nominal penarikan. Setelah itu, admin dapat mengirimkan bukti transfer untuk menyelesaikan permintaan tarik dana.

C. Pengujian

Untuk mengetahui apakah aplikasi ini berjalan dengan baik, maka dilakukannya pengujian Black Box yang mensimulasikan penggunaan aplikasi oleh pengguna akhir yang berfokus pada interaksi antarmuka dan fungsionalitasnya yang terlihat oleh pengguna. Berikut adalah hasil pengujian Black Box dari aplikasi ini:

TABEL IV
PENGUJIAN BLACK BOX FUNGSIONAL CUSTOMER

| Pengujian | Detail Pengujian | Hasil Uji |
|-------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| Registrasi | Memasukkan email, password, dan data diri | Valid |
| Unggah Foto KTP dan Foto Selfie KTP | Mengunggah foto KTP dan foto Selfie KTP pada form register | Valid |
| Login | Memasukkan email dan password | Valid |
| Melihat Vendor Katering | Menampilkan seluruh vendor sesuai kota pemesan | Valid |
| | Menampilkan seluruh vendor sesuai kota pada alamat yang dipilih | Valid |
| | Menampilkan seluruh vendor sesuai pencarian berdasarkan nama vendor atau kategori menu | Valid |
| Melihat Data dan Menu Vendor Katering | Menampilkan data dan menu yang tersedia dari vendor katering | Valid |
| Kustomisasi Menu | Memilih setidaknya satu opsi dari masing-masing kustom menu dan mengisi jumlah porsi pesanan diatas minimal pemesanan yang ditentukan oleh vendor | Valid |
| Pemesanan dan Pembayaran Katering | Memilih jadwal pengantaran minimal 1 hari sebelum acara atau kegiatan dan memilih metode pembayaran tunai atau KC Wallet | Valid |
| Penambahan Porsi | Menambahkan jumlah porsi setelah pemesanan telah dilakukan | Valid |
| Top Up KC Wallet | Memasukkan nominal sesuai minimal | Valid |
| Tarik Dana KC Wallet | Memasukkan nominal sesuai minimal dan saldo mencukupi | Valid |
| Menyelesaikan Pesanan | Menekan tombol pesanan diterima | Valid |
| Melihat Riwayat Pemesanan dan Memberikan Rating | Menampilkan riwayat pemesanan | Valid |
| | Memberikan penilaian pada pesanan yang berstatus selesai | Valid |
| | Memberikan penilaian dengan opsi satu hingga lima bintang, tidak bisa setengah bintang ataupun lebih dari lima bintang | Valid |

TABEL V
PENGUJIAN BLACK BOX FUNGSIONAL VENDOR

| Pengujian | Detail Pengujian | Hasil Uji |
|-------------------------------------|-------------------------------------------------------------|-----------|
| Registrasi | Memasukkan email, password, dan data diri | Valid |
| Unggah Foto KTP dan Foto Selfie KTP | Mengunggah foto KTP dan foto Selfie KTP pada form register | Valid |
| Login | Memasukkan email dan password | Valid |
| Mengelola Kategori Menu | Melihat, menambah, mengubah, dan menghapus kategori menu | Valid |
| Mengelola Kustom Menu | Melihat, menambah, mengubah, dan menghapus kustom menu | Valid |
| Mengelola Menu | Melihat, menambah, mengubah, dan menghapus menu | Valid |
| Melihat dan memproses pesanan masuk | Menampilkan seluruh pesanan masuk | Valid |
| | Menerima atau menolak pesanan | Valid |
| Menyelesaikan pesanan | Menyelesaikan pesanan dengan mengirim foto bukti pengiriman | Valid |
| Melihat riwayat pesanan | Menampilkan seluruh riwayat pesanan | Valid |
| Top Up KC Wallet | Memasukkan nominal sesuai minimal | Valid |
| Tarik Dana KC Wallet | Memasukkan nominal sesuai minimal dan saldo mencukupi | Valid |

TABEL VI
PENGUJIAN BLACK BOX FUNGSIONAL ADMIN

| Pengujian | Detail Pengujian | Hasil Uji |
|-----------------------------------------------------|------------------------------------------------------|-----------|
| Login | Memasukkan email dan password | Valid |
| Melihat dan memverifikasi data pemesanan dan vendor | Menerima atau menolak pendaftaran pemesan dan vendor | Valid |
| Melihat pemasukan | Menampilkan total pemasukan | Valid |
| | Menampilkan mutasi | Valid |
| Mentransfer Dana ke Pengguna | Mengirim bukti transfer | Valid |

| Pengujian | Detail Pengujian | Hasil Uji |
|---------------------------------|-------------------------------------------------------------------------------------------------------------------------------------|-----------|
| Mengatur Biaya Layanan Aplikasi | Mengatur biaya layanan aplikasi, minimal <i>top up</i> , biaya admin <i>top up</i> , minimal tarik dana, dan biaya admin tarik dana | Valid |

Hasil pengujian Black Box menunjukkan bahwa aplikasi ini berfungsi dengan baik dan sesuai dengan ekspektasi pada semua fitur yang diujikan. Pengujian dilakukan terhadap tiga peran utama yaitu customer, vendor, dan admin, dengan hasil yang konsisten valid dari pengujian. Pada peran customer yang dapat dilihat pada Tabel IV, fitur-fitur seperti registrasi, login, melihat vendor, kustomisasi menu, pemesanan dan pembayaran, menambah porsi, transaksi KC Wallet (*top up* dan tarik dana), menyelesaikan pesanan, dan memberikan penilaian semuanya berjalan tanpa masalah.

Demikian pula pada peran vendor yang dapat dilihat pada Tabel V, fungsi registrasi, login, mengelola kategori menu, mengelola kustom menu, mengelola menu, pemrosesan dan penyelesaian pesanan, dan riwayat pesanan terbukti berfungsi dengan baik. Transaksi KC Wallet untuk vendor juga berjalan dengan baik. Peran admin yang dapat dilihat pada Tabel VI juga diuji, meliputi login, verifikasi data pengguna, melihat pendapatan, transfer dana, dan pengaturan biaya layanan aplikasi. Semua fungsi admin ini dinyatakan valid oleh pengujian. Secara keseluruhan, tidak ditemukan adanya kesalahan atau ketidaksesuaian selama pengujian, yang menandakan bahwa aplikasi telah memenuhi standar fungsionalitas dan siap untuk digunakan oleh pengguna akhir.

IV. KESIMPULAN

Simpulan yang dapat diambil dari penelitian “KateringConnect: Solusi Terintegrasi untuk Vendor Katering” adalah sebagai berikut:

1. Perancangan dan pembangunan aplikasi berbasis Android untuk mewadahi proses pemesanan katering antara vendor dan pemesan berhasil dibuat. Hal ini ditandai dengan adanya beberapa fitur yang dapat memfasilitasi dalam proses pemesanan katering seperti pencarian vendor hingga pemesanan katering bagi *customer* dan penerimaan pesanan katering bagi vendor.
2. Aplikasi KateringConnect berhasil berjalan dengan baik dalam mewadahi proses pemesanan katering antara vendor dan *customer*. Hal ini dibuktikan dengan kesesuaian antara hasil yang diharapkan dari pengujian dengan hasil aktual yang didapatkan dari beberapa kali proses pengujian. Proses pengujian pada

aplikasi KateringConnect dilakukan menggunakan pengujian *Black Box*.

UCAPAN TERIMA KASIH

Puji syukur dan nikmat kepada Tuhan Yang Maha Esa atas rahmat-Nya yang melimpah sehingga penulis dapat menyelesaikan artikel ini dengan baik. Terima kasih kepada dosen pembimbing yang telah memberikan bimbingan dan masukan selama proses penulisan artikel ini. Ucapan terima kasih juga saya sampaikan kepada kedua orang tua yang telah memberikan semangat, dorongan, serta doa. Tak lupa, saya juga berterima kasih kepada teman-teman penulis yang senantiasa memberikan bantuan, dukungan, motivasi, serta doa kepada penulis.

REFERENSI

- [1] Radius Tanone and George Richard Payara, “Penerapan Firebase Realtime Database Pada Prototype Aplikasi Pemesanan Makanan Berbasis Android,” *Jurnal Teknik Informatika dan Sistem Informasi*, vol. 4, no. 3, Dec. 2018, doi: <https://doi.org/10.28932/jutisi.v4i3.870>.
- [2] M. Puspita and A. Aprilia, “FAKTOR-FAKTOR PENDORONG KONSUMEN SURABAYA MEMBELI MAKANAN DAN MINUMAN MELALUI APLIKASI GOFOOD DAN GRABFOOD,” *Jurnal Manajemen Perhotelan*, vol. 6, no. 2, pp. 88–98, Oct. 2020, doi: <https://doi.org/10.9744/jmp.6.2.88-98>.
- [3] I. Ahmad, R. I. Borman, J. Fakhrurozi, and G. G. Caksana, “Software Development Dengan Extreme Programming (XP) Pada Aplikasi Deteksi Kemiripan Judul Skripsi Berbasis Android,” *INOVTEK Polbeng - Seri Informatika*, vol. 5, no. 2, p. 297, Nov. 2020, doi: <https://doi.org/10.35314/isi.v5i2.1654>.
- [4] L. Setiyani and E. Tjandra, “ANALISIS KEBUTUHAN FUNGSIONAL APLIKASI PENANGANAN KELUHAN MAHASISWA STUDI KASUS: STMIK ROSMA KARAWANG,” *Jurnal Inovasi Pendidikan dan Teknologi Informasi (JIPTI)*, vol. 2, no. 1, pp. 8–17, Feb. 2021, doi: <https://doi.org/10.52060/pti.v2i01.465>.
- [5] A. Aulia Aziiza and A. Nur Fadhilah, “Analisis Metode Identifikasi dan Verifikasi Kebutuhan Non Fungsional,” *Applied Technology and Computing Science Journal*, vol. 3, no. 1, pp. 13–21, Sep. 2020, doi: <https://doi.org/10.33086/atcsj.v3i1.1623>.
- [6] Sheilla Riesma Intani, Asriyadi Asriyadi, and M. M. Parenreng, “Rancang Bangun Aplikasi Multi Vendor Katering Berbasis Android,” pp. 105–110, Oct. 2020.
- [7] N. Dimas, Muhammad Muharrom, None Edhi Prayitno, and None Juarni Siregar, “RANCANG BANGUN SISTEM PENERIMAAN DOKUMEN PADA PT. REASURANSI INDONESIA UTAMA,” *Jurnal Informatika Dan Teknologi Komputer*, vol. 2, no. 2, pp. 136–145, Jul. 2022, doi: <https://doi.org/10.55606/jitek.v2i2.225>.
- [8] I. S. Akbar and T. Haryanti, “Pengembangan Entity Relationship Diagram Database Toko Online Ira Surabaya,” *Computing Insight : Journal of Computer Science*, vol. 3, no. 2, pp. 28–35, Jul. 2023, doi: https://doi.org/10.30651/comp_insight.v3i2.12002.
- [9] Aziiza Yvellietchia Priyono, Guruh Aryotejo, and Satriyo Adhy, “Penerapan Metode Design Thinking untuk Perancangan Prototype Lost and Found,” *Jurnal Masyarakat Informatika/Jurnal Masyarakat Indonesia*, vol. 14, no. 2, pp. 96–107, Nov. 2023, doi: <https://doi.org/10.14710/jmasif.14.2.52662>.
- [10] Stevanu Dika Pratama, Lasimin Lasimin, and M. Noviansyah Dadaprawira, “Pengujian Black Box Testing Pada Aplikasi Edu Digital Berbasis Website Menggunakan Metode Equivalence Dan Boundary Value,” *Jurnal Teknologi Sistem Informasi dan Sistem Komputer TGD*, vol. 6, no. 2, pp. 560–560, Jul. 2023, doi: <https://doi.org/10.53513/jsk.v6i2.8166>.