

3. ANALISA DAN DESAIN SISTEM

3.1. Analisis Permasalahan

Dalam aktivitas operasional sebuah perusahaan, penyediaan konsumsi bagi karyawan merupakan salah satu aspek penting yang turut berkontribusi dalam menjaga produktivitas dan kenyamanan kerja. Perusahaan X saat ini masih menggunakan sistem pengelolaan makanan kantor yang bersifat manual. Proses-proses seperti manajemen vendor, pengelolaan menu makanan, pencatatan pengeluaran, hingga transaksi keuangan, seluruhnya belum terintegrasi dalam satu sistem yang efisien.

Berdasarkan hasil diskusi yang dilakukan secara langsung dengan salah satu karyawan yang bertugas di bagian manajemen kantin perusahaan, ternyata cukup banyak ditemukan bahwa terjadi kesalahan yang cukup fatal dan krusial, salah satunya adalah pencatatan yang tidak tertata dengan baik, serta manajemen vendor dan menu makanan yang belum terstruktur secara sistematis dan masih terpisah. Semua proses tersebut hingga saat ini masih dilakukan secara manual, seperti menggunakan buku catatan dan spreadsheet, yang rentan terhadap kesalahan pencatatan, inkonsistensi data, serta duplikasi informasi. Hal ini menyebabkan kesulitan dalam proses pemantauan, evaluasi, dan pengambilan keputusan perusahaan. Seringkali menu makanan yang muncul juga kurang sesuai dengan preferensi kesukaan mayoritas karyawan, dan tidak sedikit pula makanan yang terbuang sia-sia.

Melihat permasalahan tersebut, dibutuhkan sebuah solusi sistematis berupa sistem informasi yang dapat membantu dalam mengelola seluruh proses manajemen makanan secara menyeluruh, terintegrasi, dan efisien. Sistem ini diharapkan mampu menjadi sarana utama untuk memperbaiki proses yang selama ini berjalan secara manual dan tidak terdokumentasi dengan baik.

3.2. Analisis Kebutuhan

Berdasarkan hasil identifikasi permasalahan yang telah dipaparkan sebelumnya, maka Perusahaan X membutuhkan suatu sistem informasi manajemen makanan kantor yang terintegrasi. Sistem ini dirancang untuk menjadi platform utama dalam mengelola menu makanan, vendor penyedia konsumsi, serta pencatatan transaksi dan pengeluaran.

Beberapa kebutuhan utama sistem dapat dijabarkan sebagai berikut:

1. Manajemen Vendor dan Menu

Sistem menyediakan fitur untuk pencatatan dan pengelolaan data vendor dan menu secara detail. Informasi yang dicatat mencakup profil vendor, menu yang dimiliki, serta kedepannya adalah evaluasi kinerja vendor. Dengan adanya fitur ini, proses pemilihan dan penilaian vendor akan menjadi lebih objektif dan terdokumentasi dengan baik.

2. Manajemen Menu dengan Sistem Rekomendasi

Sistem juga dilengkapi dengan menu *planning*, dimana pihak manajemen perusahaan dapat dengan mudah mengatur jadwal pemilihan menu makanan hariannya dalam satu deck, dan sistem juga dilengkapi dengan menu *planning*, dimana pihak manajemen perusahaan dapat dengan mudah mengatur jadwal pemilihan menu makanan hariannya dalam satu deck. Dan sistem ini juga dilengkapi dengan 2 fitur dengan menggunakan sistem rekomendasi berbasis *content-based filtering*.

- Fitur Rekomendasi Menu

Fitur ini berfungsi sebagai alat bantu bagi pengguna dalam menyusun menu deck. Sistem ini memanfaatkan pendekatan *content-based filtering*, yaitu metode dalam sistem rekomendasi yang menyarankan item (dalam hal ini menu makanan) berdasarkan kemiripan atribut atau komponen dari menu yang telah dipilih sebelumnya. Misalnya, jika target menu adalah "Soto Ayam" dengan bahan seperti nasi, ayam, dan taoge, maka sistem akan mencari menu lain yang memiliki kemiripan komponennya, lalu menampilkan semua menu dari yang paling mirip hingga paling tidak mirip.

- Fitur Generate Menu

Fitur ini berfungsi untuk menyusun jadwal menu makanan mingguan atau bulanan secara otomatis. Sistem akan membuat secara otomatis mengisi deck menu berdasarkan rotasi menu yang seimbang (dimana kategori bahan utama makanan tersebut tidak boleh sama setiap harinya), dan menggunakan hasil voting historis menu sebelumnya, semakin tinggi hasil voting artinya semakin didahulukan makanan yang mirip dengan makanan tersebut. Tujuan utama dari fitur ini adalah menghemat waktu pengguna dan memastikan bahwa variasi makanan tetap terjaga, tanpa harus memilih satu per satu secara manual. Hasil

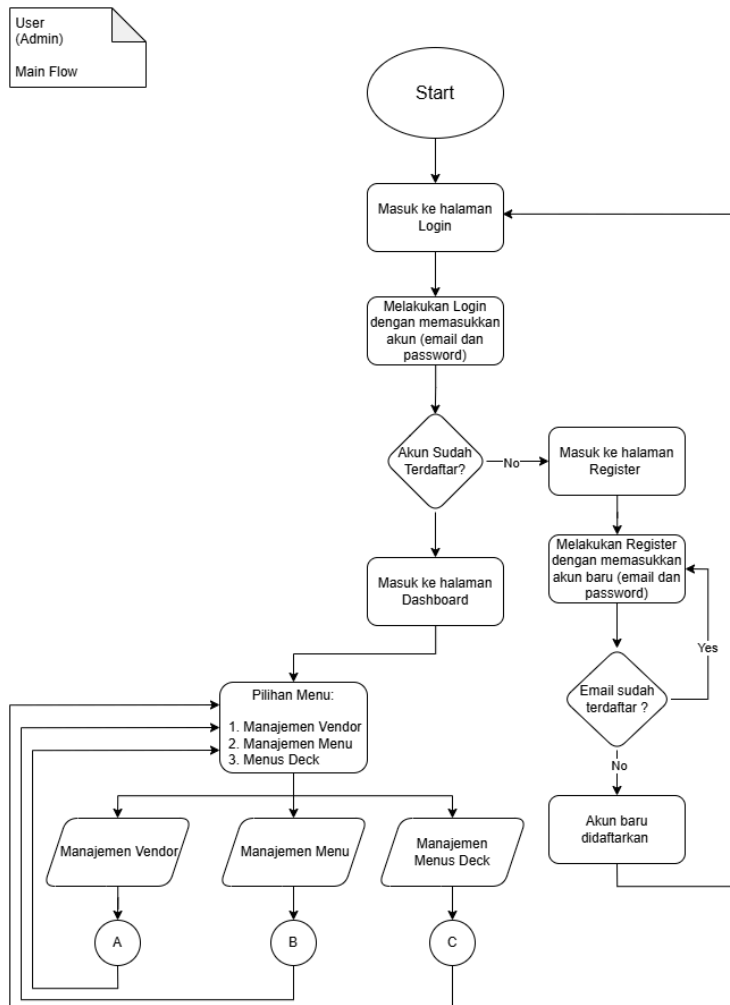
dari proses *generate* ini dapat disesuaikan kembali sebelum disimpan jika diperlukan oleh pengguna.

3. Manajemen Transaksi dan Pencatatan Biaya

Sistem ini digunakan untuk mengatur dan mendokumentasikan pelaporan biaya yang mencakup data transaksi dan pengeluaran. Data ini meliputi total pengeluaran dalam periode tertentu, histori pemesanan makanan, serta transaksi yang dilakukan. Informasi ini akan membantu dalam pelacakan anggaran dan pengambilan keputusan untuk kedepannya.

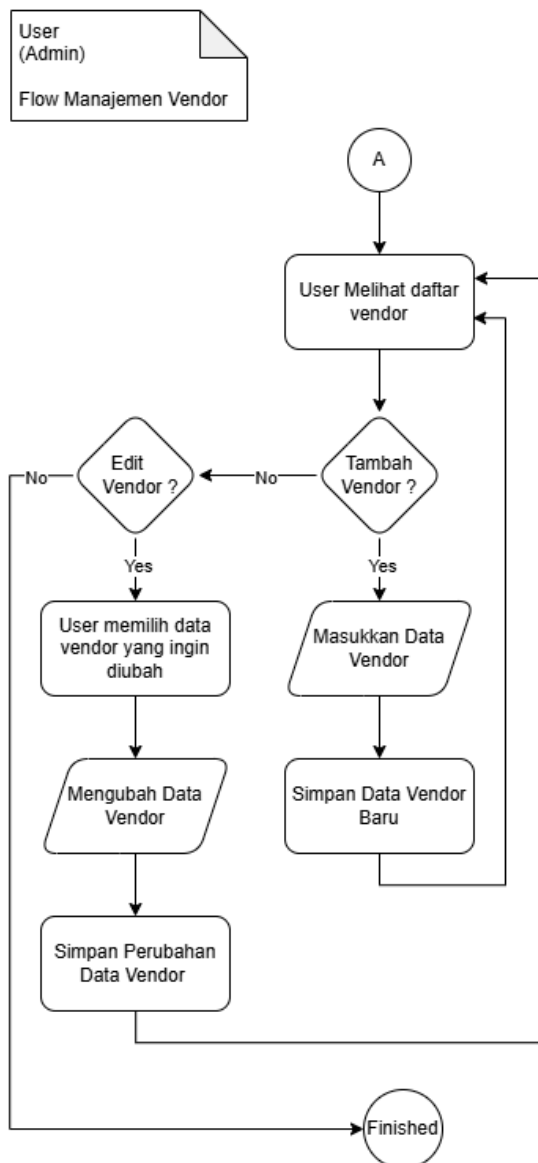
3.3. Desain Sistem

3.3.1. Flowchart



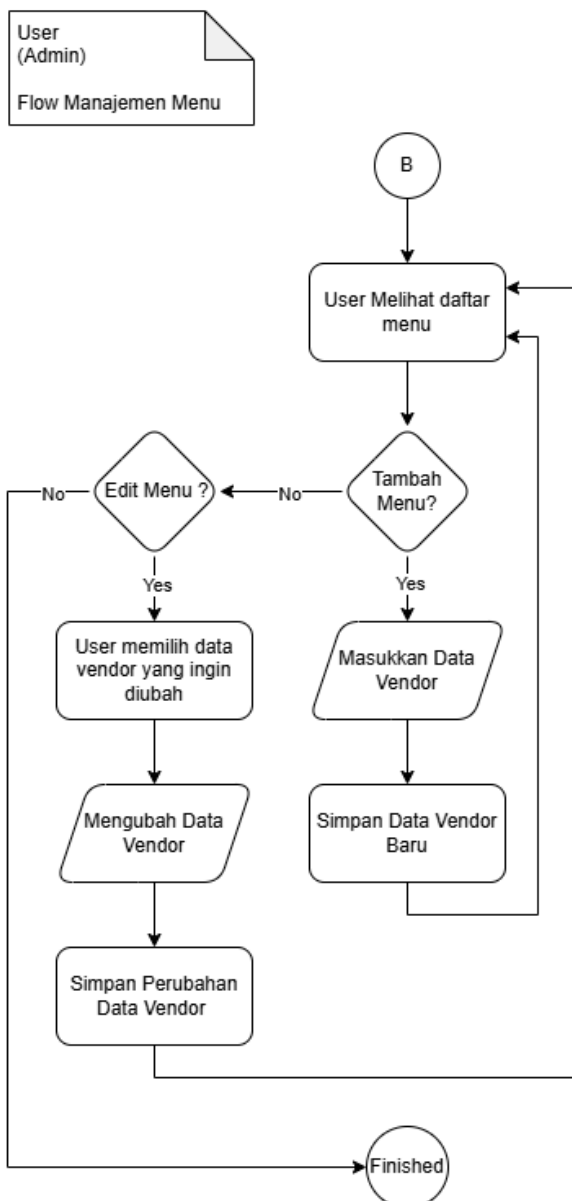
Gambar 3.1 Flowchart Main flow

Flowchart ini menggambarkan alur utama sistem aplikasi manajemen makanan kantor. Proses dimulai dari halaman login, di mana pengguna dapat masuk dengan akun yang telah terdaftar atau melakukan registrasi jika belum memiliki akun. Setelah berhasil login, pengguna akan diarahkan ke halaman dashboard yang menyediakan tiga pilihan menu utama: Manajemen Vendor, Manajemen Menu, dan Menus Deck. Setiap pilihan menu akan membawa pengguna ke proses lanjutan yang lebih detail, ditandai dengan simbol A, B, dan C untuk mengarahkan ke bagian sub-flow masing-masing menu manajemen tersebut.



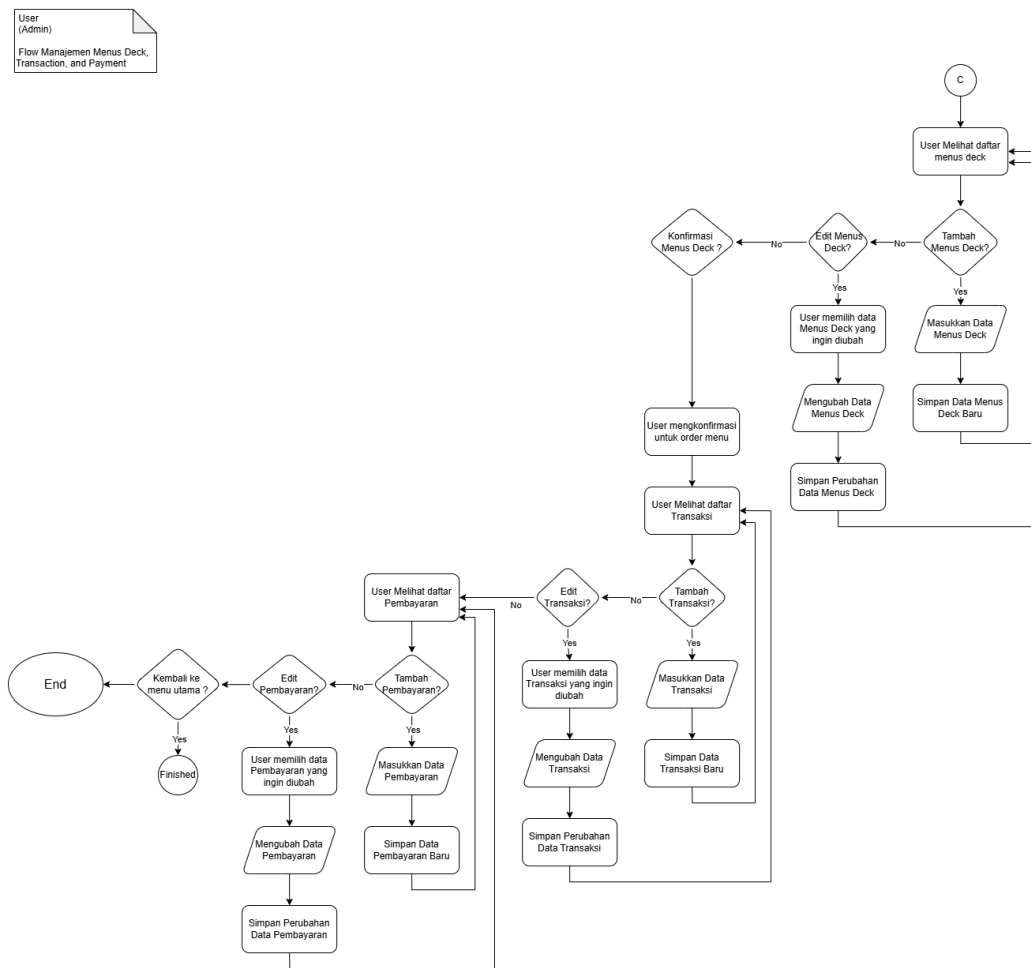
Gambar 3.2 Sub-flow Manajemen Vendor

Sub-Flowchart ini menggambarkan proses Manajemen Vendor. Proses dimulai dari titik A, di mana pengguna melihat daftar vendor yang telah ada. Pengguna dapat memilih untuk menambahkan vendor baru dengan mengisi dan menyimpan data vendor tersebut, atau mengedit vendor yang sudah ada dengan memilih data vendor yang ingin diubah, melakukan pengeditan, dan menyimpan perubahannya. Setelah salah satu aksi dilakukan, alur kembali ke daftar vendor untuk kemungkinan pengelolaan lebih lanjut, hingga akhirnya proses berakhir di titik selesai (Finished).



Gambar 3.3 Sub-flow Manajemen Menu

Flowchart ini menggambarkan alur proses Manajemen Menu. Proses dimulai dari titik B. Pengguna dapat melihat daftar menu yang tersedia dan dapat memilih untuk menambahkan menu baru dengan memasukkan serta menyimpan data menu tersebut, atau mengedit menu yang sudah ada dengan memilih data menu yang ingin diubah, lalu melakukan pengeditan dan menyimpannya. Setelah proses penambahan atau pengeditan selesai, alur kembali ke daftar menu, memungkinkan pengguna untuk terus melakukan pengelolaan menu hingga proses selesai pada titik Finished.

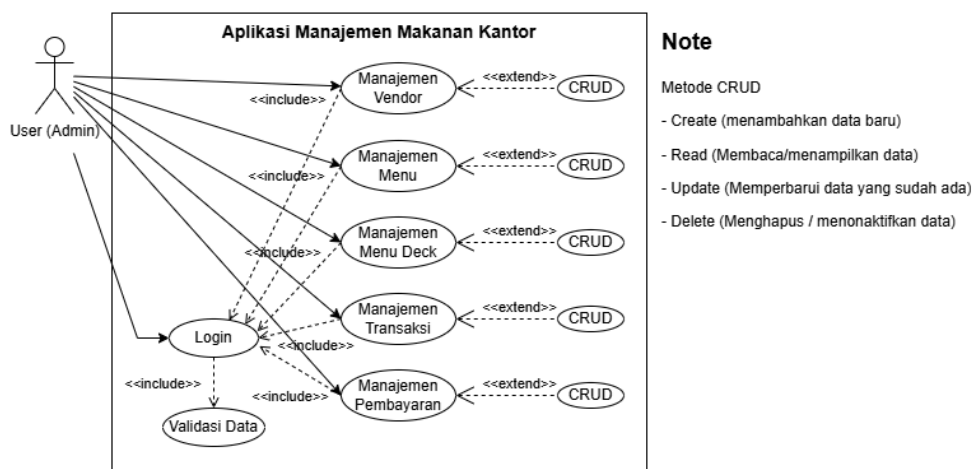


Gambar 3.4 Sub-flow Manajemen Menu Deck

Flowchart ini menggambarkan alur proses manajemen *menu deck*, transaksi, dan pembayaran. Proses dimulai dari titik C. Pengguna terlebih dahulu melihat daftar *menu deck*, lalu dapat memilih untuk menambahkan, mengedit, atau mengonfirmasi

menu deck untuk pemesanan. Setelah itu, pengguna melihat daftar transaksi dan dapat menambahkan transaksi baru, mengedit transaksi yang ada, dan menyimpan perubahannya. Selanjutnya, pengguna mengakses daftar pembayaran untuk menambahkan atau mengedit data pembayaran, kemudian menyimpan perubahan tersebut. Di akhir proses, pengguna memiliki opsi untuk kembali ke menu utama atau menyelesaikan aktivitasnya.

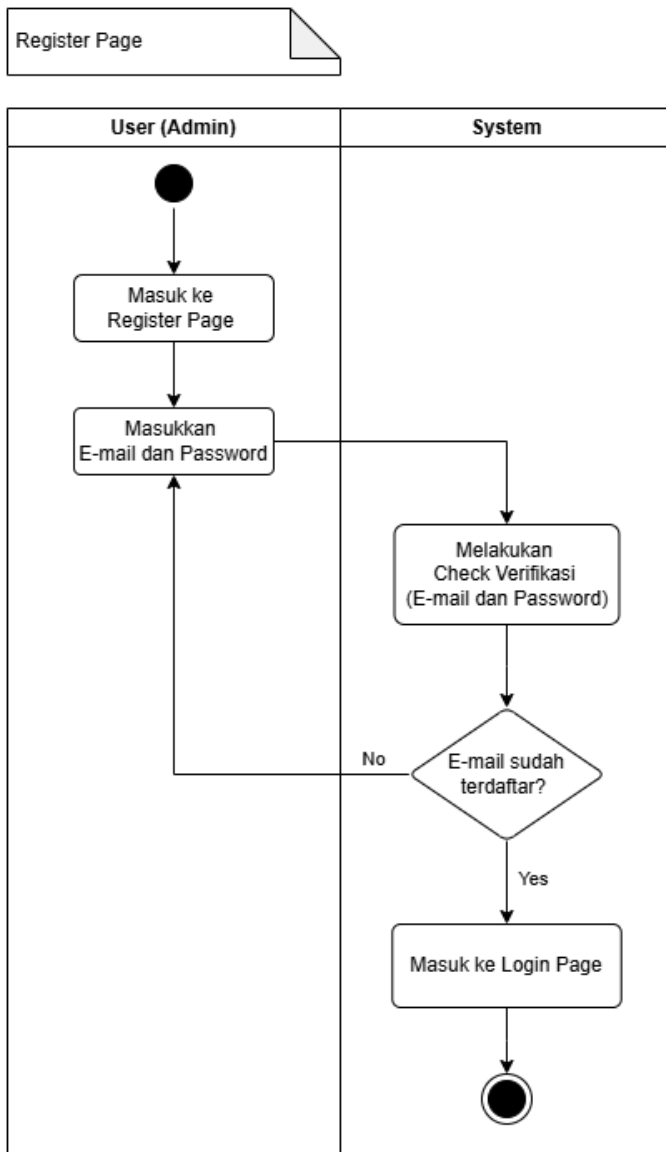
3.3.2. Use Case Diagram



Gambar 3.5 Use Case Diagram

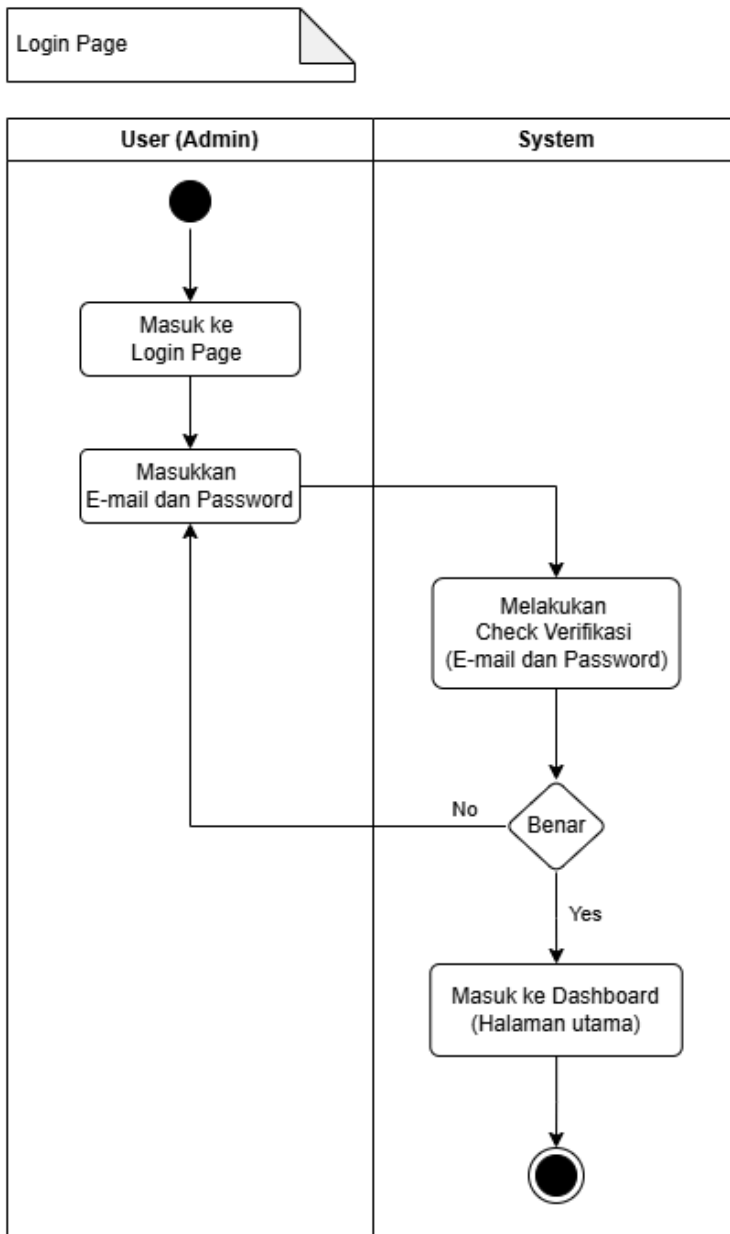
Use case diagram di atas menggambarkan alur interaksi antara pengguna (Admin) dengan sistem Aplikasi Manajemen Makanan Kantor, pengguna dapat melakukan proses Login yang mencakup Validasi Data, sebelum dapat mengakses berbagai fitur manajemen. Setelah berhasil login, pengguna bisa masuk ke fitur-fitur utama, yaitu Manajemen Vendor, Manajemen Menu, Manajemen Menu Deck, Manajemen Transaksi, dan Manajemen Pembayaran. Setiap fitur tersebut memiliki fitur CRUD (Create, Read, Update, Delete), memungkinkan pengguna untuk menambahkan, melihat, memperbaiki, atau menghapus data.

3.3.3. Activity Diagram



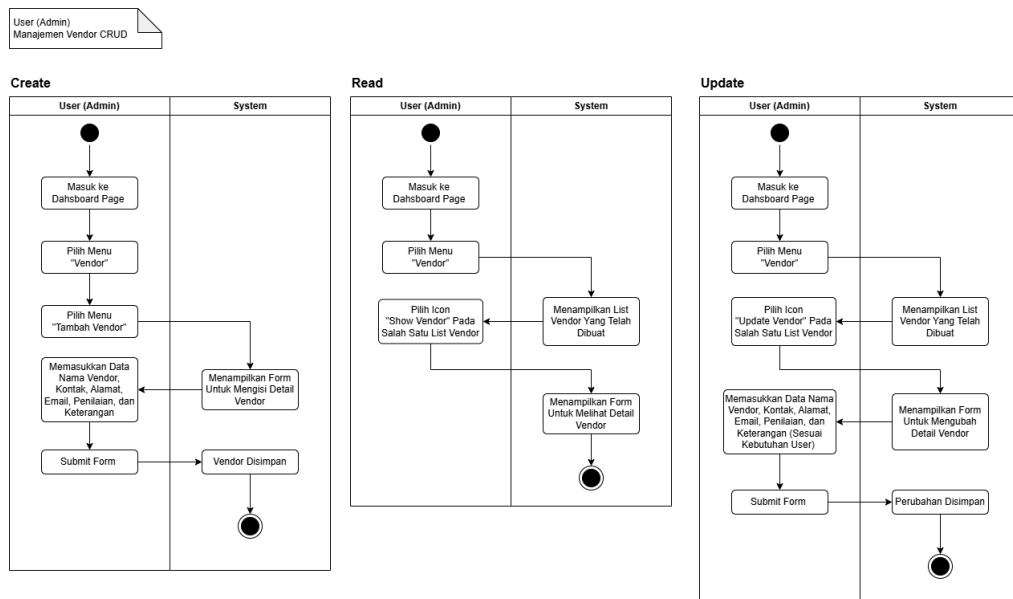
Gambar 3.6 Activity Diagram - Register Page

Register Page proses pendaftaran pengguna baru. Proses dimulai saat pengguna masuk ke halaman registrasi dan mengisi data *email* dan *password*. Sistem kemudian melakukan proses verifikasi untuk memeriksa apakah email yang dimasukkan sudah pernah terdaftar sebelumnya. Jika email tersebut sudah terdaftar, maka pengguna akan diarahkan ke halaman Login. Namun, jika email belum terdaftar, maka pengguna akan tetap berada di halaman register dan diminta kembali untuk memasukkan *email* dan *password* yang baru.



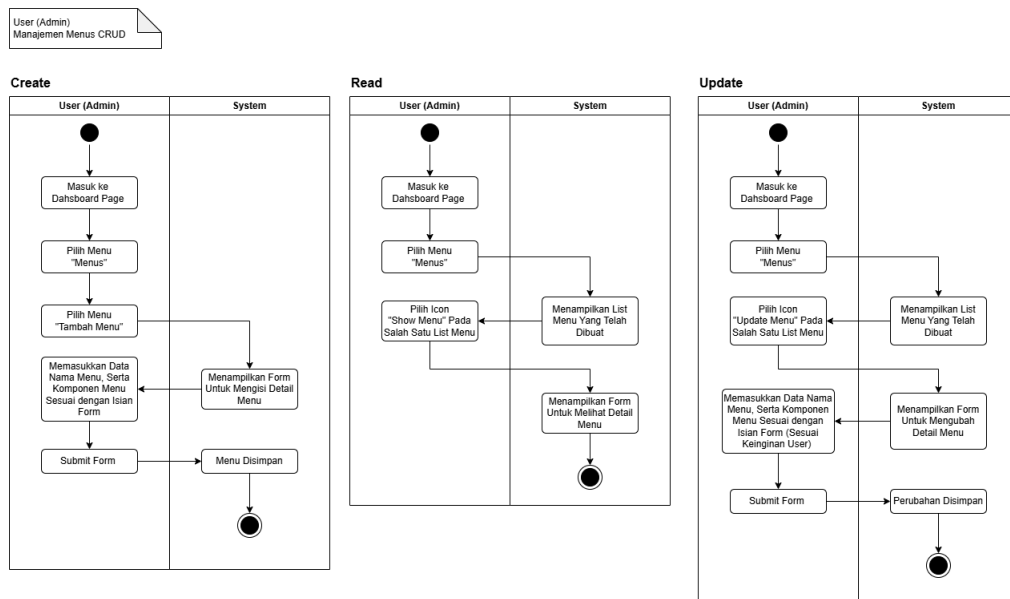
Gambar 3.7 Activity Diagram - Login Page

Login Page proses autentikasi pengguna saat mencoba masuk ke dalam sistem. Proses dimulai saat pengguna mengakses halaman *login* dan mengisi *email* dan *password*. Setelah itu, sistem melakukan proses verifikasi untuk memeriksa kecocokan data yang dimasukkan dengan data yang telah terdaftar di sistem. Jika informasi akun yang dimasukkan benar, maka pengguna akan diarahkan menuju Dashboard sebagai halaman utama aplikasi. Namun, jika informasi akun tidak sesuai, maka pengguna akan diminta kembali untuk mengisi ulang *email* dan *password* pada halaman *login*.



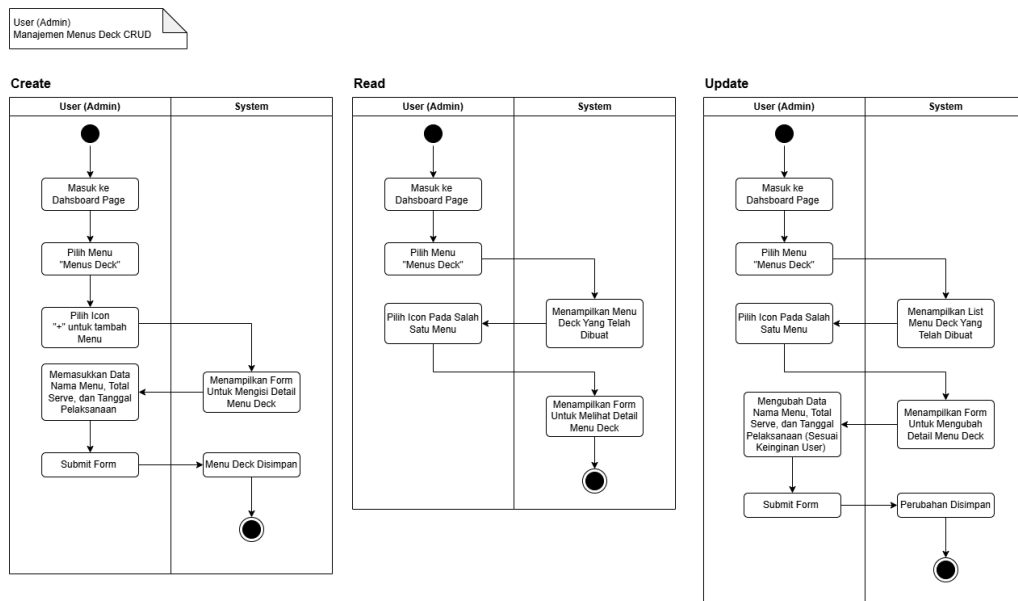
Gambar 3.8 Activity Diagram - Manajemen Vendor

Manajemen Vendor CRUD, pengguna dapat mengelola data vendor melalui proses Create, Read, dan Update. Pada proses Create, pengguna masuk ke dashboard, memilih menu Vendor, lalu memilih opsi Tambah Vendor untuk menampilkan form input data. Setelah mengisi detail vendor seperti nama, kontak, alamat, email, penilaian, dan keterangan, form disubmit dan data vendor akan disimpan ke sistem. Untuk proses Read, pengguna masuk ke menu Vendor untuk melihat daftar vendor yang telah dibuat, kemudian dapat memilih opsi Show Vendor untuk melihat detail dari vendor tertentu. Sedangkan pada proses Update, pengguna juga mengakses daftar vendor dari dashboard, memilih ikon Update Vendor untuk menampilkan form perubahan, lalu mengisi ulang atau memperbarui data vendor sesuai kebutuhan dan menyimpannya ke sistem.



Gambar 3.9 Activity Diagram - Manajemen Menu

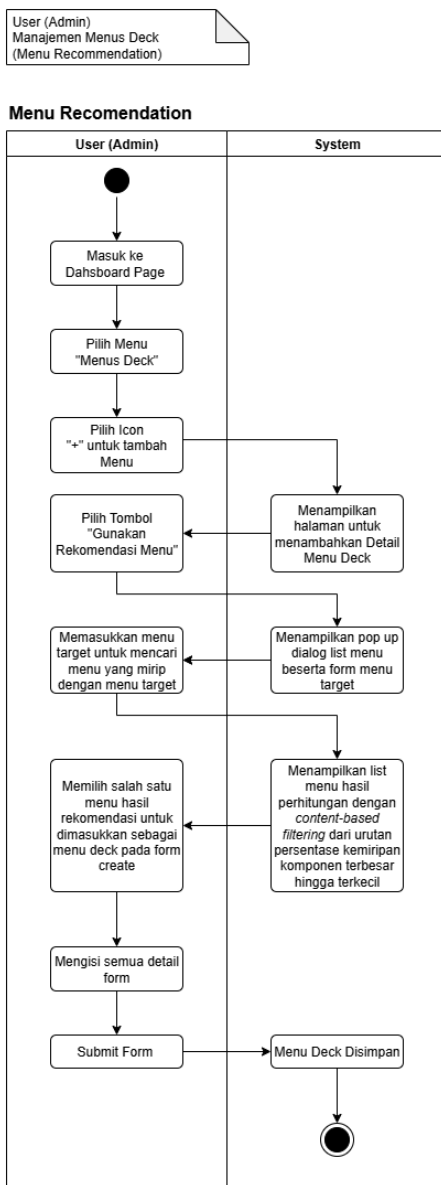
Manajemen Menu CRUD. Pengguna dapat mengelola data menu pada sistem. Pada proses Create, pengguna memulai dari halaman dashboard, memilih menu "Menus", lalu mengakses opsi "Tambah Menu" untuk menampilkan form input. pengguna mengisi nama menu beserta komponen-komponen menu (seperti karbohidrat, protein, sayur, buah, dsb) sesuai form yang tersedia, kemudian men-submit form sehingga data menu disimpan dalam sistem. Pada proses Read, pengguna dapat melihat daftar semua menu yang telah dibuat melalui menu "Menus", dan memilih ikon "Show Menu" untuk menampilkan detail dari salah satu menu. Sementara itu, proses Update diawali dengan memilih menu "Menus" dari dashboard, lalu memilih ikon "Update Menu" dari daftar yang tersedia. pengguna dapat memperbarui data nama dan komponen menu sesuai kebutuhan, kemudian menyimpan perubahan tersebut agar sistem memperbarui data menu yang bersangkutan.



Gambar 3.10 Activity Diagram - Manajemen Menu Deck

Manajemen Menu Deck CRUD. Pengguna dapat mengelola data menu deck, yaitu rencana pelaksanaan menu makanan. Pada proses Create, pengguna memulai dari halaman dashboard dan memilih menu "Menu Deck", lalu menekan ikon "+" untuk menambah data baru. Sistem akan menampilkan form untuk mengisi detail seperti nama menu, total porsi (total serve), dan tanggal pelaksanaan. Setelah data diisi dan form disubmit, sistem akan menyimpan data menu deck ke dalam database. Pada proses Read, pengguna dapat mengakses menu "Menu Deck" dari dashboard untuk melihat daftar menu deck yang sudah dibuat. Dengan memilih ikon pada salah satu menu, sistem akan menampilkan detail dari menu tersebut. Sedangkan dalam proses Update, pengguna memilih menu "Menu Deck", memilih salah satu data yang ada, lalu sistem menampilkan form detail yang bisa diubah. pengguna dapat memperbarui informasi seperti nama menu, total porsi, atau tanggal pelaksanaan sesuai kebutuhan, dan menyimpan perubahan yang telah dilakukan.

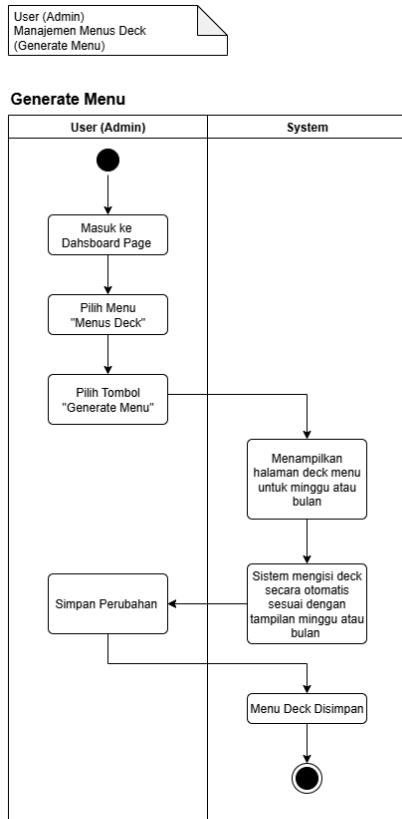
Untuk melakukan Create pada menu deck ada beberapa cara, pertama dengan menggunakan create secara manual pada halaman tersebut, kemudian ada fitur lainnya yaitu menggunakan sistem rekomendasi yang berbasis pada metode *content-based filtering*. Pertama ada fitur Rekomendasi Menu



Gambar 3.11 Activity Diagram - Manajemen Menu Deck (Menu Recommendation)

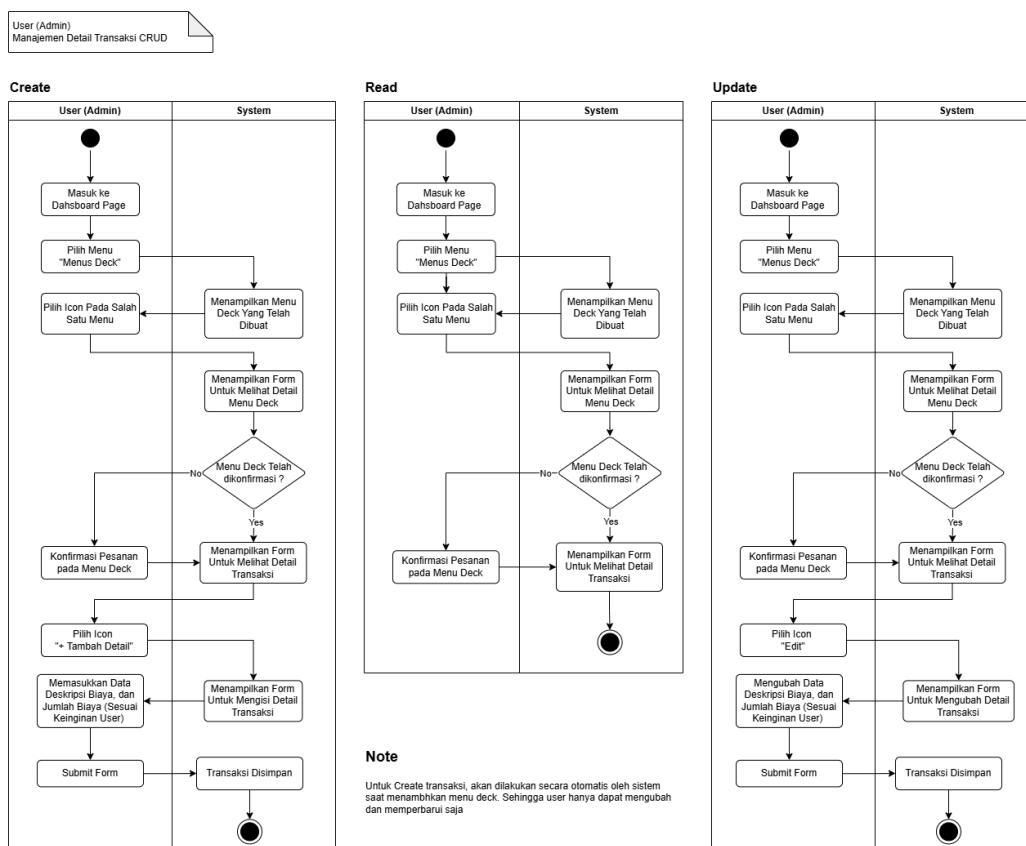
Fitur ini merupakan proses menambahkan menu ke dalam menu deck menggunakan sistem rekomendasi untuk mencari makanan yang . Proses dimulai saat pengguna masuk ke halaman dashboard, lalu memilih menu Menu Deck dan menekan ikon "+" untuk menambahkan menu baru. Setelah itu, pengguna memilih tombol "Gunakan Rekomendasi Menu", yang memicu sistem untuk menampilkan halaman input detail menu deck. Pengguna kemudian memasukkan menu target, yaitu menu yang menjadi acuan pencarian, dan sistem menampilkan pop-up dialog yang berisi list menu termasuk menu target. Sistem juga menampilkan hasil perhitungan content-based filtering, yang mengurutkan daftar menu berdasarkan tingkat kemiripan

komponen dari yang tertinggi ke terendah. pengguna memilih salah satu menu dari hasil rekomendasi, mengisi seluruh detail form, lalu menekan submit. Setelah itu, sistem menyimpan data menu deck yang telah dibuat berdasarkan menu rekomendasi tersebut.



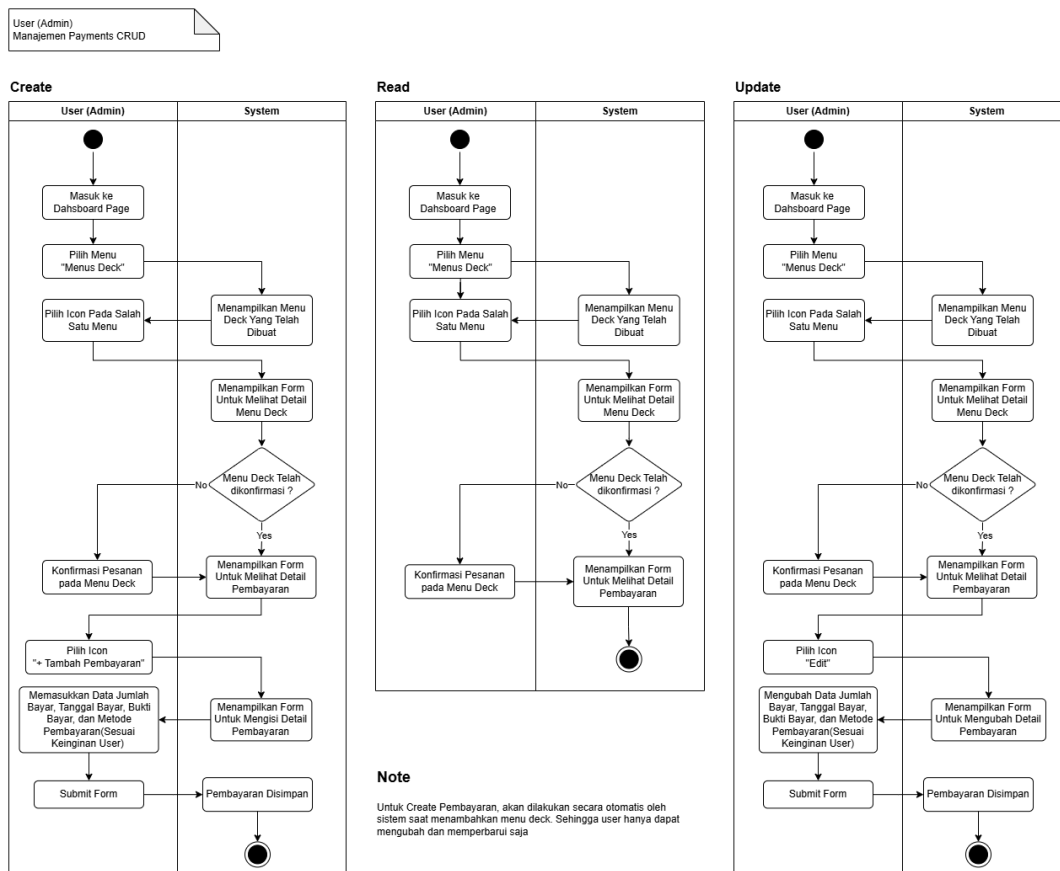
Gambar 3.12 Activity Diagram - Manajemen Menu Deck (Generate Menu)

Fitur selanjutnya adalah *Generate Menu*. Proses dimulai saat pengguna masuk ke halaman dashboard, memilih menu Menu Deck, lalu menekan tombol "Generate Menu". Sistem kemudian menampilkan halaman deck menu berdasarkan pilihan periode mingguan atau bulanan. Lalu, sistem akan secara otomatis mengisi deck menu sesuai format tampilan yang dipilih, baik mingguan maupun bulanan. Setelah itu, pengguna dapat menyimpan perubahan, dan sistem akan menyimpan menu deck tersebut.



Gambar 3.13 Activity Diagram - Manajemen Transaksi

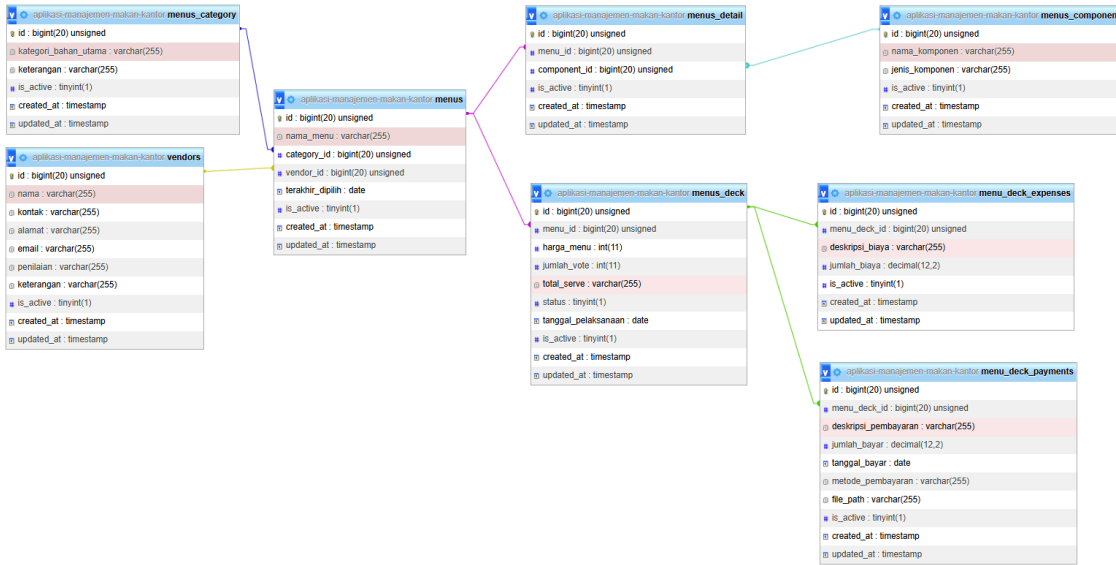
Manajemen Transaksi (CRUD) pada Menu Deck. Pengguna dapat menangani transaksi yang berkaitan dengan menu yang telah direncanakan. Pada proses Create, pengguna memulai dari Dashboard, memilih menu "Menu Deck", lalu memilih salah satu data menu untuk melihat detailnya. Sistem akan memverifikasi apakah menu deck tersebut sudah dikonfirmasi. Jika belum, pengguna perlu melakukan konfirmasi terlebih dahulu sebelum dapat melihat atau menambah transaksi. Setelah dikonfirmasi, pengguna bisa melihat detail transaksi dan memilih ikon "+ Tambah Detail" untuk mengisi deskripsi biaya serta jumlahnya. Setelah disubmit, data transaksi akan disimpan ke sistem. Untuk proses Read, alurnya mirip, dimulai dari menu Menu Deck, kemudian memilih salah satu item untuk melihat rincian menu dan transaksi—dengan pengecekan status konfirmasi terlebih dahulu. Pada proses Update, pengguna dapat memilih menu deck tertentu, membuka detail transaksi yang telah ada, lalu mengklik ikon Edit. Formulir perubahan akan muncul, memungkinkan pengguna memperbarui deskripsi atau jumlah biaya, lalu menyimpan perubahan.



Gambar 3.14 Activity Diagram - Manajemen Pembayaran

Manajemen Pembayaran (CRUD) pada Menu Deck pengguna dapat melakukan pencatatan setiap transaksi yang ada. Proses ini bermula pada tahap CREATE, pengguna memulai dari Dashboard, memilih menu "Menu Deck", lalu membuka salah satu menu deck. Sistem akan mengecek apakah menu tersebut sudah dikonfirmasi. Jika belum, pengguna harus melakukan konfirmasi terlebih dahulu. Setelah konfirmasi, sistem menampilkan form detail pembayaran dan pengguna dapat menambahkan data pembayaran baru dengan mengisi Jumlah Bayar, Tanggal Bayar, Bukti Bayar, dan Metode Pembayaran. Setelah disubmit, data pembayaran disimpan. Pada proses READ, alur serupa dilakukan tanpa pengisian data: pengguna dapat melihat detail pembayaran dari menu yang telah dikonfirmasi. Untuk UPDATE, pengguna mengikuti alur yang sama hingga melihat detail pembayaran, lalu memilih ikon Edit untuk mengubah data yang diinginkan. Setelah perubahan dilakukan dan form disubmit, sistem akan memperbarui data pembayaran.

3.3.4. Entity Relationship Diagram (ERD)



Gambar 3.15 Entity Relationship Diagram (ERD)

ERD ini menggambarkan bagaimana alur sistem manajemen makanan kantor bekerja. Dimulai dari tabel vendors yang menyediakan berbagai Menu. Dan Setiap menu memiliki category_id yang merujuk ke tabel menu_category untuk pengelompokan berdasarkan bahan utama, dan detail komponen penyusun menu seperti karbohidrat, protein, sayur, atau buah diatur dalam tabel master menu_component yang terhubung melalui tabel menu_detail. Untuk tabel menu_deck, akan menyimpan informasi seperti menu yang dipilih menu_id, total serve, harga menu saat itu, jumlah vote saat itu, status konfirmasi, dan tanggal pelaksanaan. Untuk pencatatan pembayaran dan biaya, disimpan dalam tabel menu_deck yang sekaligus memiliki relasi ke tabel menu_deck_expenses untuk mencatat deskripsi dan jumlah biaya-biaya terkait, serta ke menu_deck_payments yang mencatat detail pembayaran seperti deskripsi, jumlah yang dibayar, tanggal, metode pembayaran, hingga file path bukti pembayaran, memastikan semua transaksi tercatat dengan lengkap.

3.3.5. Desain Tabel Database

Tabel 3.1 Tabel vendors

Nama	Atribut	Tipe Data	Keterangan
id	PK	BIGINT	Primary key
nama	NOT NULL	VARCHAR	Nama vendor
kontak	NOT NULL	VARCHAR	Informasi kontak vendor
alamat	NOT NULL	VARCHAR	Alamat vendor
email	NOT NULL	VARCHAR	Email vendor
penilaian	NOT NULL	VARCHAR	Penilaian vendor
keterangan	NULLABLE	VARCHAR	Deskripsi atau keterangan tambahan vendor
is_active	NOT NULL	BOOLEAN	Status aktif (default: true)
created_at	NULLABLE	DATETIME	Waktu pembuatan record
updated_at	NULLABLE	DATETIME	Waktu terakhir record diperbarui

Tabel vendors menyimpan data seluruh penyedia makanan (vendor). Informasi yang disimpan mencakup nama, kontak, alamat, email, status aktif vendor, serta penilaian dan deskripsi tambahan jika tersedia. Tabel ini berfungsi sebagai sumber utama penyedia layanan makanan yang akan ditautkan ke berbagai menu makanan yang mereka sediakan.

Tabel 3.2 Tabel menus

Nama	Atribut	Tipe Data	Keterangan
id	PK, NOT NULL	UUID / INT	Primary key
nama_menu	NOT NULL	VARCHAR	Nama menu
karbohidrat	NULLABLE	FLOAT	Kandungan karbohidrat
protein	NULLABLE	FLOAT	Kandungan protein
sayur	NULLABLE	FLOAT	Kandungan sayur
buah	NULLABLE	FLOAT	Kandungan buah
keterangan	NULLABLE	TEXT	Deskripsi menu
kategori_bahan_utama	NULLABLE	VARCHAR	Misalnya: Nasi, Mie, Daging
is_active	NOT NULL	BOOLEAN	Status aktif
vendor_id	FK, NOT NULL	UUID / INT	Relasi ke tabel Vendors
harga	NOT NULL	DECIMAL	Harga menu
jumlah_vote	NULLABLE	INT	Jumlah total vote

terakhir_dipilih	NULLABLE	DATE / DATETIME	Terakhir kali menu dipilih
------------------	----------	-----------------	----------------------------

Tabel `menus` berfungsi untuk menyimpan daftar semua menu makanan yang tersedia. Setiap menu memiliki id unik, `nama_menu`, harga, serta informasi kapan terakhir kali menu tersebut dipilih. Tabel ini juga berelasi dengan tabel `vendors` untuk menunjukkan vendor penyedia menu dan `menus_category` untuk mengelompokkan menu berdasarkan kategori bahan utama, serta memiliki kolom `is_active`.

Tabel 3.3 Tabel `menus_detail`

Nama	Atribut	Tipe Data	Keterangan
Nama	Atribut	Tipe Data	Keterangan
id	PK	BIGINT	Primary key
menu_id	FK, NOT NULL	BIGINT	Foreign key ke tabel <code>menus</code>
component_id	FK, NOT NULL	BIGINT	Foreign key ke tabel <code>menus_component</code>
is_active	NOT NULL	BOOLEAN	Status aktif (default: true)
created_at	NULLABLE	DATETIME	Waktu pembuatan record
updated_at	NULLABLE	DATETIME	Waktu terakhir record diperbarui

Tabel `menus_detail` berfungsi sebagai tabel penghubung (pivot table) antara `menus` dan `menus_component`, yang memungkinkan satu menu memiliki banyak komponen dan satu komponen dapat menjadi bagian dari banyak menu. Ini merinci komponen-komponen yang membentuk suatu menu, seperti karbohidrat, protein, sayur, atau buah, dengan kolom `is_active` untuk mengelola status ketersediaan detail komponen menu.

Tabel 3.4 Tabel `menus_category`

Nama	Atribut	Tipe Data	Keterangan
Nama	Atribut	Tipe Data	Keterangan
id	PK	BIGINT	Primary key
kategori_bahan_utama	NOT NULL	VARCHAR	Kategori bahan utama
keterangan	NULLABLE	VARCHAR	Deskripsi atau keterangan kategori
is_active	NOT NULL	BOOLEAN	Status aktif (default: true)
created_at	NULLABLE	DATETIME	Waktu pembuatan record
updated_at	NULLABLE	DATETIME	Waktu terakhir record diperbarui

Tabel `menus_category` digunakan untuk mengelompokkan menu berdasarkan kategori bahan utamanya, seperti Nasi, Mie, atau Daging. Setiap kategori memiliki id unik, `kategori_bahan_utama` sebagai nama kategori, keterangan opsional untuk deskripsi lebih lanjut, dan `is_active` untuk mengelola status aktif kategori tersebut.

Tabel 3.5 Tabel `menus_component`

Nama	Atribut	Tipe Data	Keterangan
Nama	Atribut	Tipe Data	Keterangan
id	PK	BIGINT	Primary key
nama_komponen	NOT NULL	VARCHAR	Nama komponen (misalnya: Tahu, Telur)
jenis_komponen	NOT NULL	VARCHAR	Jenis komponen (misalnya: karbohidrat, protein, sayur, buah)
is_active	NOT NULL	BOOLEAN	Status aktif (default: true)
created_at	NULLABLE	DATETIME	Waktu pembuatan record
updated_at	NULLABLE	DATETIME	Waktu terakhir record diperbarui

Tabel `menus_component` menyimpan daftar komponen-komponen yang dapat menyusun sebuah menu, seperti karbohidrat, protein, sayur, atau buah. Setiap komponen memiliki id unik, `nama_komponen` (misalnya "Nasi Putih", "Ayam Bakar"), dan `jenis_komponen` yang mengklasifikasikan komponen tersebut, serta `is_active` untuk mengelola status aktif komponen.

Tabel 3.6 Tabel `menus_deck`

Nama	Atribut	Tipe Data	Keterangan
id	PK	BIGINT	Primary key
menu_id	FK, NOT NULL	BIGINT	Foreign key ke tabel <code>menus</code>
total_serve	NULLABLE	VARCHAR	Jumlah total sajian
harga_menu	NULLABLE	INTEGER	Harga menu
jumlah_vote	NULLABLE	INTEGER	Jumlah total vote
status	NOT NULL	BOOLEAN	Status konfirmasi (default: false)
tanggal_pelaksanaan	NULLABLE	DATE	Tanggal pelaksanaan
is_active	NOT NULL	BOOLEAN	Status aktif (default: true)
created_at	NULLABLE	DATETIME	Waktu pembuatan record
updated_at	NULLABLE	DATETIME	Waktu terakhir record diperbarui

Tabel `menu_deck` berfungsi untuk mencatat setiap kali sebuah menu dipilih atau direncanakan untuk periode tertentu. Informasi yang disimpan meliputi `menu_id` yang dipilih, total serve, harga menu, jumlah vote yang diterima, status konfirmasi, dan tanggal pelaksanaan menu tersebut, serta status aktif `is_active` untuk catatan rencana menu.

Tabel 3.7 Tabel `menu_deck_expenses`

Nama	Atribut	Tipe Data	Keterangan
<code>id</code>	PK	BIGINT	Primary key
<code>menu_deck_id</code>	FK, NOT NULL	BIGINT	Foreign key ke tabel <code>menu_deck</code>
<code>deskripsi_biaya</code>	NOT NULL	VARCHAR	Deskripsi biaya
<code>jumlah_biaya</code>	NOT NULL	DECIMAL(12, 2)	Jumlah biaya
<code>is_active</code>	NOT NULL	BOOLEAN	Status aktif (default: true)
<code>created_at</code>	NULLABLE	DATETIME	Waktu pembuatan record
<code>updated_at</code>	NULLABLE	DATETIME	Waktu terakhir record diperbarui

Tabel `menu_deck_payments` mencatat rincian pembayaran yang terkait dengan setiap `menu_deck`. Informasi yang disimpan meliputi `deskripsi_pembayaran`, `jumlah_bayar` yang akurat dengan dua desimal, `tanggal_bayar`, `metode_pembayaran` (misalnya Transfer atau Tunai), serta `file_path` opsional untuk bukti pembayaran, dan `is_active` untuk status catatan pembayaran.

Tabel 3.8 Tabel `menu_deck_payments`

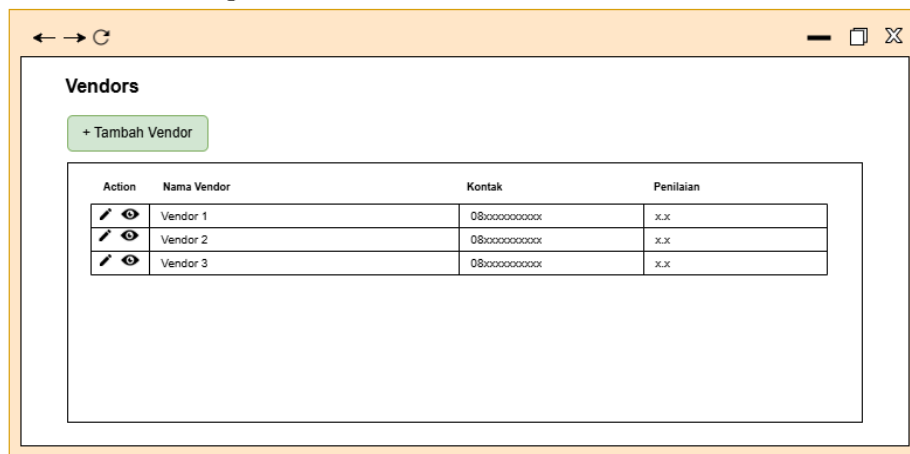
Nama	Atribut	Tipe Data	Keterangan
<code>id</code>	PK	BIGINT	Primary key
<code>menu_deck_id</code>	FK, NOT NULL	BIGINT	Foreign key ke tabel <code>menu_deck</code>
<code>deskripsi_pembayaran</code>	NOT NULL	VARCHAR	Deskripsi pembayaran
<code>jumlah_bayar</code>	NOT NULL	DECIMAL(12, 2)	Jumlah yang dibayarkan
<code>tanggal_bayar</code>	NOT NULL	DATE	Tanggal pembayaran
<code>metode_pembayaran</code>	NOT NULL	VARCHAR	Metode pembayaran (misalnya: Transfer, Cash)
<code>file_path</code>	NULLABLE	VARCHAR	Path file bukti pembayaran
<code>is_active</code>	NOT NULL	BOOLEAN	Status aktif (default: true)

created_at	NULLABLE	DATETIME	Waktu pembuatan record
updated_at	NULLABLE	DATETIME	Waktu terakhir record diperbarui

Tabel menu_deck_payments mencatat rincian pembayaran yang terkait dengan setiap menus_deck. Informasi yang disimpan meliputi deskripsi_pembayaran, jumlah_bayar yang akurat dengan dua desimal, tanggal_bayar, metode_pembayaran (misalnya Transfer atau Tunai), serta file_path opsional untuk bukti pembayaran, dan is_active untuk status catatan pembayaran.

3.3.6. UI Design

Vendors - Main Page



Gambar 3.16 UI Design Vendor (Main Page)

Halaman ini menampilkan daftar seluruh vendor yang telah ditambahkan atau terdaftar sebelumnya, lengkap dengan detail seperti nama vendor, kontak, dan penilaian vendor. Pada halaman ini juga Terdapat tombol “+ Tambah Vendor” yang dapat digunakan untuk menambahkan vendor baru dan akan mengarahkan pengguna ke halaman Vendor - *Create Page*. Setiap baris data vendor dilengkapi dengan tombol “edit” dan “show”, yang masing-masing berfungsi untuk mengarahkan pengguna ke halaman *Edit Page* dan *Show Page*.

Vendors - Create Page

The image displays two screenshots of a web form titled "Vendors - Create Page".

The top screenshot shows the form with the following fields:

- Nama Vendor
- Kontak
- Alamat
- E-mail

The bottom screenshot shows the form with the following fields and buttons:

- Penilaian
- Keterangan
- Save
- Cancel

Gambar 3.17 UI Design Vendor (Create Page)

Halaman ini menampilkan *form* untuk menambahkan vendor baru. Pengguna diminta mengisi detail vendor seperti nama, kontak, dan alamat. Untuk isian seperti e-mail, penilaian, dan keterangan bersifat opsional. Setelah semua data diisi, pengguna dapat menekan tombol "Save" untuk menyimpan data ke dalam database.

Vendors - Show Page

The image shows two browser window screenshots of the 'Vendors - Show Page' UI design. Both windows have a title bar with navigation and window control icons.

The top window displays a form titled 'Vendors' with the following fields:

- Nama Vendor: Vendor X
- Kontak: 08xxxxxxxx
- Alamat: Alamat X
- E-mail: vendorx@gmail.com

The bottom window displays a form titled 'Vendors' with the following fields and a button:

- Penilaian: x.x
- Keterangan: Keterangan Vendor X
- Back button

Gambar 3.18 UI Design Vendor (Show Page)

Halaman ini hanya untuk menampilkan *form* yang berisi detail vendor, pengguna dapat melihat keseluruhan detail vendor mulai dari nama, kontak, dan alamat, *e-mail*, penilaian, dan keterangan. Disini pengguna hanya dapat melihat detailnya saja, dan tidak dapat melakukan perubahan data yang ada.

Vendors - Edit Page

The image displays two screenshots of a web application's 'Vendors - Edit Page'. The top screenshot shows a form with the following fields:

- Nama Vendor:** Vendor X
- Kontak:** 08xxxxxxxxxx
- Alamat:** Alamat X
- E-mail:** vendorx@gmail.com

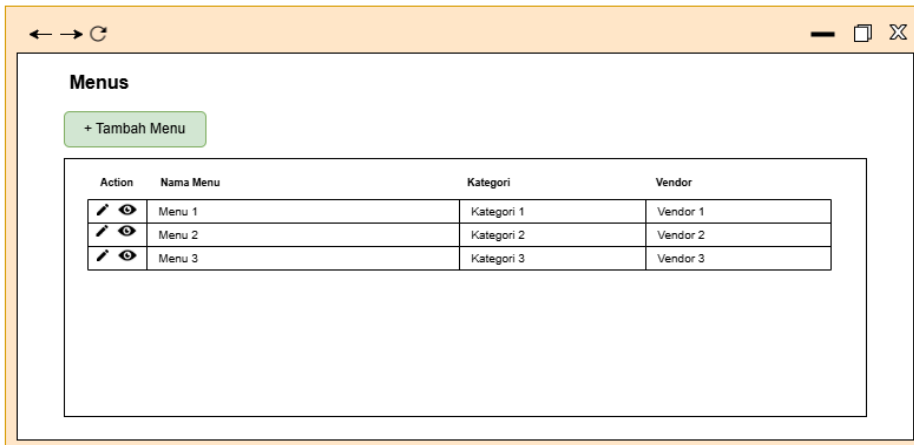
The bottom screenshot shows a form with the following fields and buttons:

- Penilaian:** x.x
- Keterangan:** Keterangan Vendor X
- Buttons:** Save (blue), Cancel (red)

Gambar 3.19 UI Design Vendor (Edit Page)

Halaman ini menampilkan *form* yang berisi detail vendor yang telah ada, pengguna dapat melihat dan melakukan perubahan pada data vendor mulai dari nama, kontak, dan alamat, *e-mail*, penilaian, dan keterangan. Setelah selesai melakukan perubahan, pengguna dapat menekan tombol “*save*” untuk menyimpan pembaruan ke dalam *database*.

Menus - Main Page



Gambar 3.20 UI Design Menus (Main Page)

Halaman ini menampilkan daftar seluruh menu yang telah ditambahkan atau terdaftar sebelumnya, lengkap dengan detail seperti nama menu, kategori, dan vendor penyediaannya. Pada halaman ini juga Terdapat tombol “+ Tambah Menu” yang dapat digunakan untuk menambahkan menu baru dan akan mengarahkan pengguna ke halaman Menu - *Create Page*. Setiap baris data menu dilengkapi dengan tombol “edit” dan “show”, yang masing-masing berfungsi untuk mengarahkan pengguna ke halaman *Edit Page* dan *Show Page*.

Menus - Create Page

The image displays two screenshots of a web application window titled "Menus - Create Page".

The top screenshot shows a form with the following fields:

- Nama Menu
- Karbohidrat
- Protein
- Sayur

The bottom screenshot shows the same form with additional fields and buttons:

- Buah
- Kategori Bahan Utama
- Harga
- Vendor
- Save button
- Cancel button

Gambar 3.21 UI Design Menus (Create Page)

Halaman ini menampilkan *form* untuk menambahkan menu baru. Pengguna diminta mengisi detail komponen menu seperti nama, karbohidrat, protein, sayur, buah, kategori bahan utama, dan juga detail menu lainnya seperti harga dan nama vendor. Setelah semua data diisi, pengguna dapat menekan tombol "Save" untuk menyimpan data ke dalam database.

Menus - Show Page

The image displays two screenshots of a web application interface for the 'Menus - Show Page'. Both screenshots are contained within a light orange window frame with standard navigation icons (back, forward, refresh) and window controls (minimize, maximize, close).

The top screenshot shows a form titled 'Menus' with the following fields:

- Nama Menu:** Input field containing 'Menu x'.
- Karbohidrat:** Input field containing 'Menu x'.
- Protein:** Input field containing 'Item x'.
- Sayur:** Input field containing 'Item x'.

The bottom screenshot shows a more detailed form titled 'Menus' with the following fields and elements:

- Buah:** Input field containing 'Item x'.
- Kategori Bahan Utama:** Input field containing 'Kategori x'.
- Harga:** Input field containing 'Rp. .xx.xxx'.
- Vendor:** Dropdown menu containing 'Vendor x'.
- Status:** A checked checkbox.
- Back:** A red button.

Gambar 3.22 UI Design Menus (Show Page)

Halaman ini hanya untuk menampilkan *form* yang berisi detail menu, pengguna dapat melihat keseluruhan detail komponen menu mulai dari nama, karbohidrat, protein, sayur, buah, kategori bahan utama, hingga detail menu lainnya seperti harga dan nama vendor. Disini pengguna hanya dapat melihat detailnya saja, dan tidak dapat melakukan perubahan data yang ada.

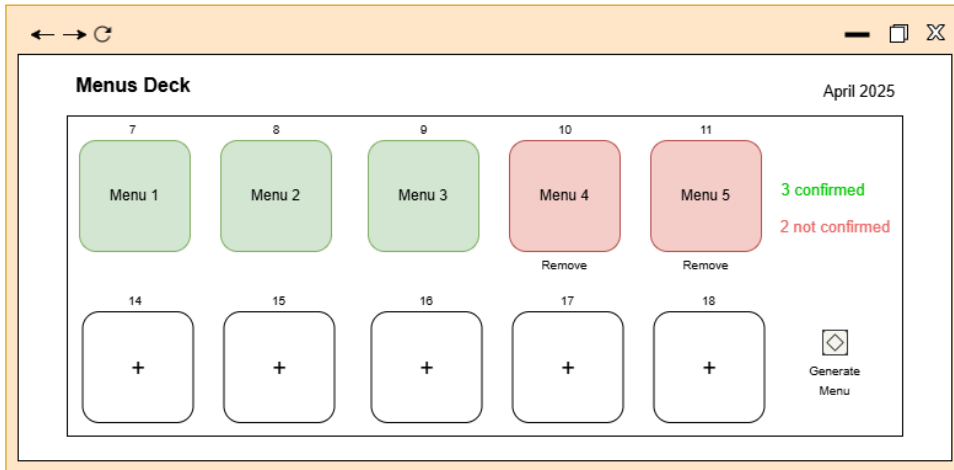
Menus - Edit Page

The image displays two screenshots of a web application's 'Menus - Edit Page'. The top screenshot shows a form with four input fields: 'Nama Menu' (Menu x), 'Karbohidrat' (Menu x), 'Protein' (Item x), and 'Sayur' (Item x). The bottom screenshot shows a more detailed form with fields for 'Buah' (Item x), 'Kategori Bahan Utama' (Kategori x), 'Harga' (Rp. xx.xxx), 'Vendor' (Vendor x), and a 'Status' checkbox (checked). It also includes 'Save' and 'Cancel' buttons.

Gambar 3.23 UI Design Menus (Edit Page)

Halaman ini menampilkan *form* yang berisi detail menu yang telah ada, pengguna dapat melihat dan melakukan perubahan pada data komponen menu mulai dari nama, karbohidrat, protein, sayur, buah, kategori bahan utama, hingga detail menu lainnya seperti harga, nama vendor, dan juga status menu. Setelah selesai melakukan perubahan, pengguna dapat menekan tombol “save” untuk menyimpan pembaruan ke dalam *database*.

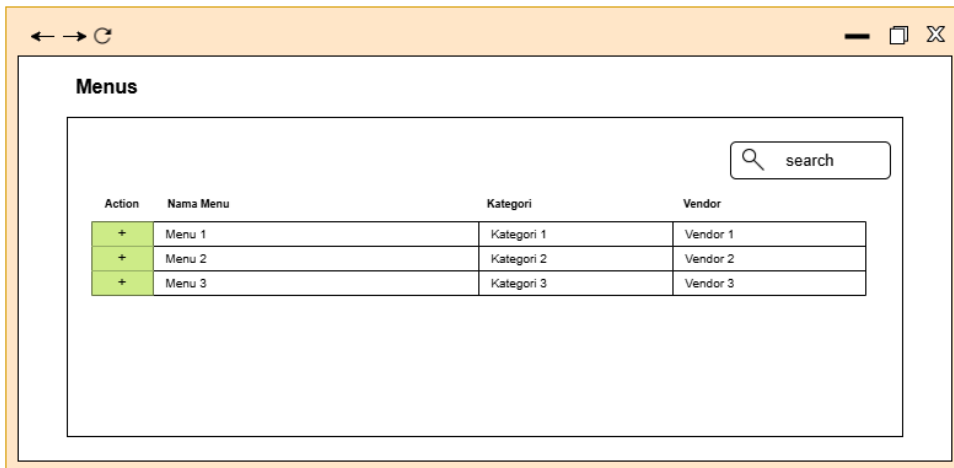
Menus Deck - Main Page



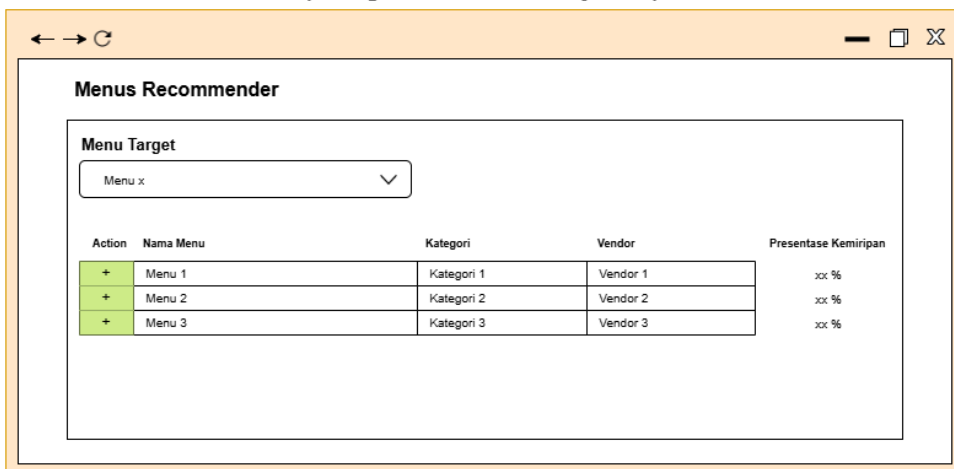
Gambar 3.24 UI Design Menus Deck (MainPage)

Halaman ini menampilkan daftar menu makanan untuk satu minggu berdasarkan tanggal. Setiap kotak mewakili satu hari, yang bisa diisi dengan menu tertentu. Kotak berwarna hijau menandakan menu sudah *confirmed*, sementara warna merah menunjukkan menu belum *confirmed*. Pengguna dapat menambahkan menu baru ke hari yang kosong dengan menekan tombol "+", atau menghapus menu yang belum dikonfirmasi. Di sisi kanan terdapat statistik jumlah menu yang sudah dikonfirmasi atau belum, serta tombol "*Generate Menu*" untuk membuat menu otomatis.

Menus Deck - Add Menu (Manually)



Menus Deck - Add Menu (Using Recommender System)



Gambar 3.25 UI Design Menus Deck (Create Page)

Add Menu (Manually)

Halaman ini memungkinkan pengguna untuk menambahkan menu secara manual ke dalam *Menus Deck*. Tersedia daftar menu lengkap beserta kategori dan vendor penyedia. Pengguna dapat menelusuri daftar menggunakan kolom pencarian, lalu menambahkan menu yang diinginkan dengan menekan tombol “+” di kolom Action.

Add Menu (Using Recommender System)

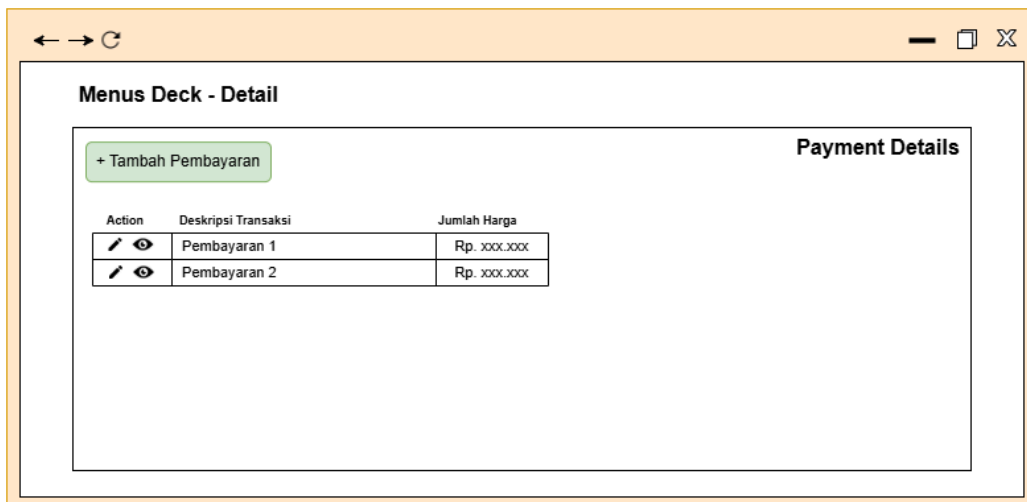
Halaman ini sama fungsinya untuk menambahkan menu ke dalam *Menus Deck*, tetapi dengan bantuan sistem rekomendasi. Pengguna dapat memilih Menu Target yang akan dijadikan acuan, lalu sistem akan menampilkan daftar menu serupa beserta

kategori, vendor, dan persentase kemiripan. Menu yang diinginkan bisa langsung ditambahkan melalui tombol "+". Fitur ini bertujuan untuk mempercepat proses penjadwalan menu dengan opsi-opsi yang relevan dan sesuai preferensi sebelumnya.

The screenshot shows a web application window titled "Menus Deck - Detail". At the top left is a "+ Tambah Transaksi" button. Below it, there are three input fields: "Total Menu", "Harga Menu", and "Total harga". To the right of these is a "Transaction Details" section with a "Status" dropdown menu containing the options "Fully Paid / Half Paid / Not Paid". Below the main input fields, there are two rows of "Deskripsi Biaya" and "Total harga" fields, each with an edit icon and a delete "x" button. At the bottom, there is a "Total Semua" field.

Gambar 3.26 UI Design Menus Deck (Show Page) 1

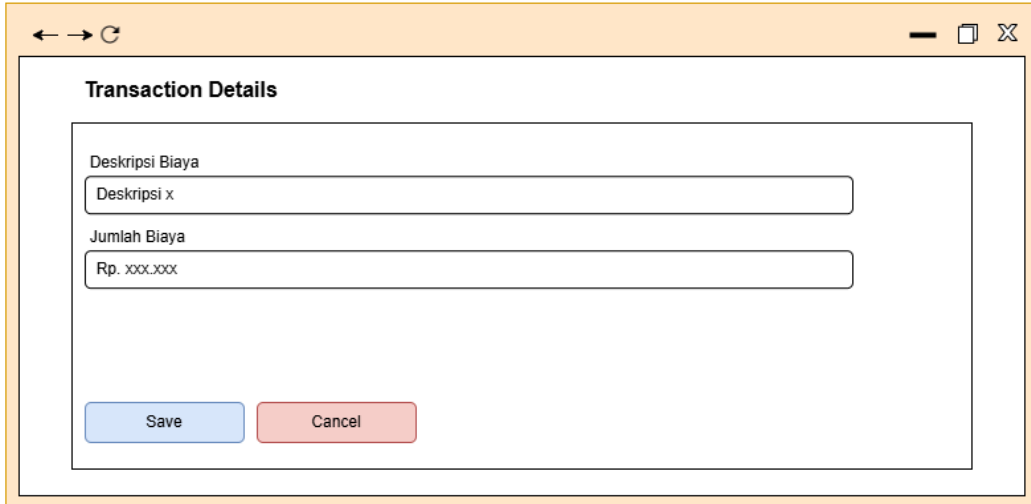
Halaman ini masih tetap di halaman yang sama, tetapi ini merupakan lanjutan setelah "Menus Deck", fitur ini digunakan untuk mencatat perincian biaya. Di bagian atas terdapat tombol "+ Tambah Transaksi" yang memungkinkan pengguna menambahkan rincian biaya baru. Kemudian di bawahnya memuat *form* dengan isian jumlah menu, harga per menu, dan otomatis menghitung total harga. Di bawahnya terdapat kolom untuk memasukkan deskripsi biaya tambahan lain serta total harga masing-masing, yang dapat diedit atau dihapus menggunakan ikon "edit" dan tombol "x". Total harga keseluruhan ditampilkan di bagian bawah. Di sisi kanan halaman, terdapat status pembayaran yang menunjukkan kondisi transaksi, yaitu apakah sudah "Fully Paid", "Half Paid", atau "Not Paid".



Gambar 3.27 UI Design Menus Deck (Show Page) 2

Halaman ini merupakan lanjutan setelah "*Transaction Details*" digunakan untuk mencatat rincian pembayaran atau transaksi. Di bagian atas terdapat tombol "+ Tambah Pembayaran" yang memungkinkan pengguna menambahkan pembayaran baru. Di bawahnya, terdapat tabel berisi Deskripsi Transaksi, dan Jumlah Harga. Kolom "Action" menyediakan ikon untuk melihat dan mengedit data, sementara kolom lainnya menampilkan deskripsi transaksi beserta jumlah nominal pembayaran.

Menus Deck - Add Transaction Details



The image shows a web browser window titled "Menus Deck - Add Transaction Details". The browser's address bar contains navigation icons (back, forward, refresh) and window control icons (minimize, maximize, close). The main content area is titled "Transaction Details" and contains a form with two input fields. The first field is labeled "Deskripsi Biaya" and contains the text "Deskripsi x". The second field is labeled "Jumlah Biaya" and contains the text "Rp. xxx.xxx". At the bottom of the form, there are two buttons: a blue "Save" button and a red "Cancel" button.

Gambar 3.28 UI Design Transaction Details (Create Page)

Halaman ini menampilkan *form* untuk menambahkan "*Transaction Details*" baru. Pengguna diminta untuk mengisi detail rincian biaya seperti deskripsi dan jumlah biaya. Setelah semua data diisi, pengguna dapat menekan tombol "Save" untuk menyimpan data ke dalam database.

The image shows a web browser window with a title bar containing navigation icons (back, forward, refresh) and window control icons (minimize, maximize, close). The main content area is titled "Payment Details" and contains a form with the following elements:

- Jumlah Bayar:** A text input field containing "Rp. xxx.xxx".
- Tanggal Bayar:** A text input field containing "xx-xx-xxxx".
- Bukti Bayar:** A file upload area with a "Choose File" button and the text "No File Chosen".
- Metode Pembayaran:** A text input field containing "Metode x".
- Buttons:** A blue "Save" button and a red "Cancel" button are positioned at the bottom of the form.

Gambar 3.29 UI Design Payment Details (Create Page)

Halaman ini menampilkan *form* untuk menambahkan "*Payment Details*" baru. Pengguna diminta untuk mengisi detail rincian transaksi mulai dari jumlah bayar, tanggal, bukti pembayaran (memungkinkan pengguna mengunggah file sebagai bukti pembayaran), hingga metode pembayaran (memilih metode yang digunakan, misalnya transfer, tunai, atau lainnya). Setelah semua data diisi, pengguna dapat menekan tombol "Save" untuk menyimpan data ke dalam database.