

Product Backlog Proyek RPL (Metode Scrum)

Kelompok 6 (5 C Sistem Informasi)

1. Jihan Nabillah (701230022)
2. Dwi Astina (701230030)
3. Sinta Ulan Dari (701230178)

Nama Proyek: Pool Snack – Sistem Pemesanan Snack Arena Billiard

Product Owner: Kelompok 11 (User)

Tanggal Dibuat: 25 September 2025

Deskripsi Singkat : Berisi tentang deskripsi sistem pemesanan snack atau minuman berbasis web yang terhubung dengan QR code unik per meja billiard. Customer bisa memesan langsung dari meja, kasir menerima order real-time, dan admin mengelola menu. Sistem ini bertujuan meningkatkan kenyamanan customer, mempercepat transaksi, dan memudahkan manajemen kasir.

Bagian 1: Daftar User Stories

Berikut adalah daftar semua fitur yang diinginkan untuk produk ini, diurutkan berdasarkan prioritas. *User story* baru akan terus ditambahkan ke dalam *backlog* ini.

ID User Story	User Story	Prioritas	Estimasi (Story Point)	Status	Sprint
US001	Sebagai customer, saya ingin registrasi & membuat akun agar dapat login.	Sangat Tinggi	5	Done	Sprint 1
US002	Sebagai customer, saya ingin login agar aktivitas saya tercatat.	Sangat Tinggi	5	Done	Sprint 1

US003	Sebagai customer, saya ingin logout agar sesi saya aman setelah digunakan.	Tinggi	3	Done	Sprint 1
US004	Sebagai sistem, saya ingin menghapus isi keranjang setelah logout.	Sangat Tinggi	4	Done	Sprint 2
US005	Sebagai sistem, saya ingin memastikan setiap user memiliki session unik agar data tidak tercampur.	Sangat Tinggi	8	Done	Sprint 1
US006	Sebagai customer, saya ingin scan QR code umum agar diarahkan ke halaman login.	Sangat Tinggi	3	Done	Sprint 1
US007	Sebagai customer, saya ingin memilih nomor meja setelah login agar pesanan tercatat.	Sangat Tinggi	5	Done	Sprint 1
US008	Sebagai sistem, saya ingin memvalidasi nomor meja agar tidak digunakan user lain.	Sangat Tinggi	5	Done	Sprint 1
US009	Sebagai customer, saya ingin melihat daftar menu (gambar, harga, kategori).	Tinggi	5	Done	Sprint 1
US010	Sebagai customer, saya ingin menambahkan item ke keranjang.	Sangat Tinggi	8	Done	Sprint 1
US011	Sebagai sistem, saya ingin menghitung total harga otomatis di keranjang.	Sangat Tinggi	3	Done	Sprint 1
US012	Sebagai customer, saya ingin mengubah jumlah item di keranjang.	Tinggi	3	Done	Sprint 1
US013	Sebagai customer, saya ingin menghapus item dari keranjang.	Tinggi	3	Done	Sprint 1
US014	Sebagai customer, saya ingin checkout & memilih metode pembayaran.	Sangat Tinggi	8	Done	Sprint 2
US015	Sebagai customer, saya ingin upload bukti pembayaran QRIS.	Tinggi	5	Done	Sprint 2

US016	Sebagai kasir, saya ingin login ke dashboard.	Sangat Tinggi	5	Done	Sprint 3
US017	Sebagai kasir, saya ingin menerima order secara real-time.	Sangat Tinggi	8	Done	Sprint 3
US018	Sebagai kasir, saya ingin mengubah status order.	Sangat Tinggi	5	Done	Sprint 3
US019	Sebagai admin, saya ingin login agar dapat mengelola menu.	Tinggi	3	Done	Sprint 4
US020	Sebagai admin, saya ingin menambah menu (Create).	Tinggi	5	Done	Sprint 4
US021	Sebagai admin, saya ingin mengedit menu (Update).	Tinggi	5	Done	Sprint 4
US022	Sebagai admin, saya ingin melihat daftar menu (Read).	Sedang	3	Done	Sprint 4
US023	Sebagai admin, saya ingin menghapus menu (Delete).	Sedang	5	Done	Sprint 4
US024	Sebagai kasir, saya ingin memverifikasi pembayaran tunai/QRIS.	Tinggi	5	Done	Sprint 3
US025	Sebagai kasir, saya ingin input pesanan manual.	Tinggi	5	Done	Sprint 3
US026	Sebagai sistem, saya ingin mengirim notifikasi status pesanan ke customer.	Tinggi	8	Done	Sprint 2
US027	Sebagai sistem, saya ingin mengirim notifikasi ke kasir bila customer upload bukti pembayaran.	Tinggi	3	Done	Sprint 2
US028	Sebagai customer, saya ingin melihat riwayat pesanan hari ini.	Sedang	3	Done	Sprint 5
US029	Sebagai customer, saya ingin melihat nota digital.	Tinggi	3	Done	Sprint 5
US030	Sebagai kasir, saya ingin melihat laporan transaksi harian.	Tinggi	8	Done	Sprint 5
US031	Sebagai kasir, saya ingin export laporan harian PDF/Excel.	Tinggi	5	Done	Sprint 5
US032	Sebagai kasir, saya ingin melihat laporan mingguan.	Sedang	8	Done	Sprint 5
US033	Sebagai kasir, saya ingin export laporan mingguan PDF/Excel.	Sedang	5	Done	Sprint 5

US034	Sebagai kasir, saya ingin melihat menu terlaris.	Sedang	5	Done	Sprint 5
--------------	--	--------	---	------	----------

Bagian 2: Acceptance Criteria (AC)

untuk User Story Prioritas Tinggi Acceptance Criteria (AC) adalah checklist yang mendefinisikan kapan sebuah user story dianggap "Selesai" (Done). Jika semua poin di AC terpenuhi, maka user story tersebut berhasil diimplementasikan.

AC untuk US001 : Registrasi Customer

Skenario : Customer ingin membuat akun baru agar dapat login ke sistem.

- ☐ Customer dapat mengisi nama, email/username, password.
 - ☐ Sistem memvalidasi format email.
 - ☐ Password minimal 6 karakter.
 - ☐ Email tidak boleh duplikat di database.
 - ☐ Jika semua valid → akun berhasil dibuat.
 - ☐ Password disimpan menggunakan hashing (bukan plaintext).
-

AC untuk US002 : Login Customer

Skenario : Customer ingin login menggunakan kredensial akun.

- ☐ Customer dapat memasukkan email/username dan password.
- ☐ Jika kredensial valid → customer masuk ke halaman pilih meja / menu.
- ☐ Jika kredensial tidak valid → sistem menampilkan pesan "Email atau password salah."
- ☐ Password tersimpan dengan hashing/enkripsi (tidak plaintext).
- ☐ Session customer dibuat setelah login.
- ☐ Session hanya berlaku untuk user yang sedang login (isolasi session antar pengguna).
- ☐ Jika tombol "Ingat Saya" dipilih → session diperpanjang sesuai durasi.

AC untuk US003 : Logout Customer

Skenario: Customer ingin keluar dari akun dengan aman.

- ☐ Customer dapat menekan tombol “Logout”. ·
 - ☐ Sistem mengakhiri session login customer. ·
 - ☐ Customer diarahkan kembali ke halaman login. ·
 - ☐ Data keranjang terhapus otomatis setelah logout. ·
 - ☐ Nomor meja yang aktif pada session tersebut dibersihkan. ·
 - ☐ Semua akses halaman setelah logout tidak bisa dibuka tanpa login ulang.
-

AC untuk US004 : Clear Cart Setelah Logout

Skenario: Sistem membersihkan keranjang customer yang logout.

- ☐ Saat logout, isi keranjang user dihapus.
- ☐ Keranjang tidak terbawa saat login kembali.
- ☐ Keranjang tidak bercampur dengan user lain.

AC untuk US005 : Unique Session per User

Skenario: Sistem menjaga agar session setiap customer berbeda.

- [] Setiap login menghasilkan session unik yang tidak berbagi data dengan user lain.
 - [] Sistem mengikat keranjang pada session, bukan pada device.
 - [] Jika customer A logout → session ditutup dan keranjang terhapus.
 - [] Jika customer B login di device yang sama → keranjang harus kosong.
 - [] Tidak ada data keranjang, meja, atau order customer lain yang muncul di akun lain.
 - [] Jika session kadaluarsa → user harus login ulang dan keranjang hilang.
-

AC untuk US006 : Scan QR Code Umum

Skenario: Customer scan QR di meja untuk membuka halaman login.

- [] QR dapat discan dari kamera HP.
 - [] QR mengarah ke URL login sistem.
 - [] QR tidak langsung masuk tanpa login.
-

AC untuk US007 : Pilih Nomor Meja

Skenario: Customer memilih meja setelah login.

- [] Setelah login, customer wajib memilih nomor meja sebelum masuk ke halaman menu.
- [] Daftar meja muncul berupa dropdown atau list.
- [] Meja yang sedang dipakai customer lain tidak muncul / tidak dapat dipilih.
- [] Jika meja belum dipilih, sistem tidak mengizinkan melanjutkan ke menu.

- [] Nomor meja tersimpan dalam session customer.
 - [] Nomor meja tampil pada ringkasan pesanan dan dashboard kasir.
-

AC untuk US008 : Validasi Nomor Meja

Skenario: Sistem mengecek apakah meja sedang digunakan.

- [] Sistem menolak jika meja sudah dipakai pengguna lain.
 - [] Pesan error muncul saat nomor tidak valid.
 - [] Meja hanya aktif untuk satu session customer.
-

AC untuk US009 : Lihat Daftar Menu

Skenario: Customer ingin melihat menu untuk memesan.

- [] Menu tampil dalam bentuk list atau grid.
 - [] Setiap menu menampilkan gambar, nama, deskripsi singkat, harga, dan kategori.
 - [] Customer bisa melakukan filter berdasarkan kategori (makanan/minuman).
 - [] Jika menu kosong → sistem menampilkan pesan “Belum ada menu tersedia.”
 - [] Perubahan menu oleh admin (create/update/delete) langsung ter-update di halaman customer tanpa reload manual. •
 - [] Tampilan responsif untuk mobile, tablet, dan desktop.
-

AC untuk US010 : Tambah Item ke Keranjang

Skenario: Customer memilih menu dan menambahkannya ke keranjang.

- ☐ Customer dapat menekan tombol “Tambah”.
 - ☐ Item masuk ke keranjang sesuai jumlah.
 - ☐ Keranjang menampilkan total item.
 - ☐ Sistem menolak jika stock habis (jika ada cek stock).
-

AC untuk US011: Auto Total Harga

Skenario: Sistem menghitung total harga secara otomatis

- ☐ Setiap item memiliki harga satuan.
 - ☐ Total harga = $\text{sum}(\text{item_qty} \times \text{harga})$.
 - ☐ Total berubah real-time saat qty berubah.
 - ☐ Tidak boleh ada perhitungan manual.
-

AC untuk US012 : Ubah Jumlah Item

Skenario: Customer mengubah qty item dalam keranjang.

- ☐ Customer bisa tambah/kurang qty.
 - ☐ Qty minimal 1.
 - ☐ Total harga otomatis ter-update.
-

AC untuk US013: Hapus Item dari Keranjang

Skenario: Customer ingin menghapus item dari keranjang.

- ☐ Ada tombol hapus pada setiap item.
 - ☐ Setelah dihapus, total harga ter-update.
 - ☐ Tidak ada item “hantu” tertinggal.
-

AC untuk US014: Checkout & Pilih Metode Pembayaran

Skenario: Customer menyelesaikan pesanan.

- ☐ Customer dapat memilih pembayaran Tunai atau QRIS.
 - ☐ Jika Tunai → tampil instruksi bayar ke kasir.
 - ☐ Jika QRIS → tampil QR code pembayaran.
 - ☐ Setelah checkout → order tercatat sebagai “Baru”.
-

AC untuk US015: Upload Bukti Pembayaran QRIS

Skenario: Customer bayar via QRIS lalu upload bukti.

- ☐ Customer dapat memilih file foto.
 - ☐ Bukti tersimpan ke folder /uploads/.
 - ☐ Notifikasi ke kasir dibuat (jika fitur ada).
 - ☐ Order berubah status “Menunggu Verifikasi”.
-

AC untuk US016 : Login Kasir

Skenario: Kasir masuk dashboard.

- [] Kasir memasukkan username/password.
 - [] Jika kredensial valid → masuk ke dashboard kasir.
 - [] Jika tidak valid → tampil pesan “Username atau password salah.”
 - [] Password kasir tersimpan dengan hashing/enkripsi.
 - [] Session kasir dibuat saat login.
 - [] Session hanya memberi akses ke fitur kasir (role-based access control).
-

AC untuk US017: Order Masuk Real-Time

Skenario: Kasir menerima pesanan baru.

- [] Order baru muncul tanpa reload (AJAX/Pusher).
 - [] Order baru muncul dalam waktu kurang dari 2 detik setelah customer checkout.
 - [] Kasir dapat melihat detail pesanan (menu, jumlah, total, metode pembayaran).
 - [] Order ditampilkan dengan status “Baru”.
 - [] Jika koneksi putus → order tetap tidak hilang.
-

AC untuk US018: Update Status Order

Skenario: Kasir memproses order.

- [] Kasir dapat mengubah status order menggunakan tombol aksi.
 - [] Status yang valid: Baru → Diproses → Selesai (tidak bisa skip tahap).
 - [] Setiap perubahan status tersimpan di database.
 - [] Customer mendapatkan notifikasi real-time ketika status berubah.
 - [] Sistem menolak perubahan status jika order sudah “Selesai”
-

AC untuk US019: Login Admin

Skenario: Admin masuk halaman manajemen menu

- [] Admin memasukkan email dan password.
 - [] Sistem memvalidasi email unik & format benar.
 - [] Autentikasi hanya untuk role “admin”.
 - [] Jika berhasil, admin diarahkan ke dashboard manajemen menu.
 - [] Setelah logout, admin tidak bisa akses halaman dashboard.
-

AC untuk US020: Admin Tambah Menu (Create)

Skenario: Admin menambahkan menu.

- [] Form input berisi nama menu, harga, kategori, foto, dan ketersediaan.
 - [] Validasi wajib: nama, harga, kategori tidak boleh kosong.
 - [] Foto hanya menerima format JPG/PNG.
 - [] Menu baru tersimpan ke database.
 - [] Menu langsung muncul di daftar menu admin & halaman customer.
-

AC untuk US021: Admin Edit Menu (Update)

Skenario: Admin mengedit menu

- [] Admin dapat mengubah nama menu, harga, kategori, deskripsi, dan status tersedia/tidak.
 - [] Sistem menolak harga tidak valid (angka negatif atau huruf).
 - [] Jika foto diganti, foto lama terhapus dari server.
 - [] Perubahan data tersimpan ke database.
 - [] Perubahan langsung tercermin di daftar menu customer.
-

AC untuk US022: Admin Lihat Daftar Menu (Read)

Skenario: Admin melihat daftar menu

- [] Sistem menampilkan semua menu dalam bentuk tabel/list.
 - [] Setiap item menampilkan nama, harga, foto, status.
 - [] Menu dapat difilter berdasarkan kategori.
 - [] Menu dapat dicari menggunakan kolom search.
 - [] Data ditampilkan sesuai data real di database.
-

AC untuk US023: Admin Hapus Menu (Delete)

Skenario: Admin menghapus menu

- [] Admin dapat menghapus menu melalui tombol “Delete”.
 - [] Sistem menampilkan popup konfirmasi sebelum menghapus.
 - [] Jika dihapus, data menu hilang dari database.
 - [] Foto terkait juga terhapus dari server.
 - [] Menu yang dihapus tidak lagi tampil di customer & kasir.
-

AC untuk US024 : Verifikasi Pembayaran Tunai/QRIS oleh Kasir

Skenario: Kasir memverifikasi pembayaran customer agar transaksi tercatat valid.

- [] Kasir dapat membuka detail pesanan yang berstatus “Menunggu Konfirmasi”.
 - [] Untuk pembayaran tunai → kasir dapat klik tombol “Konfirmasi Pembayaran Tunai”.
 - [] Untuk pembayaran QRIS → bukti pembayaran yang di-upload customer harus dapat dilihat kasir.
 - [] Jika bukti tidak jelas/invalid → kasir dapat menolak dengan alasan.
 - [] Setelah verifikasi → status berubah menjadi “Pembayaran Diterima”.
 - [] Sistem mencatat waktu verifikasi dan nama kasir.
 - [] Order yang sudah diverifikasi tidak dapat diedit kembali.
 - [] Dashboard kasir melakukan refresh data secara real-time.
-

AC untuk US025 : Input Pesanan Manual oleh Kasir

Skenario: Kasir membuat pesanan tanpa scan QR (customer langsung pesan di kasir).

- [] Kasir dapat memilih menu dari daftar menu admin.
 - [] Kasir dapat menambah kuantitas tiap item.
 - [] Kasir dapat memasukkan nomor meja secara manual.
 - [] Kasir dapat memilih metode pembayaran (Tunai/QRIS).
 - [] Jika memilih QRIS → kasir bisa upload bukti pembayaran customer.
 - [] Order manual ditandai dengan label “Dibuat oleh Kasir”.
 - [] Order muncul di dashboard kasir dan admin seperti order normal.
-

AC untuk US026 : Notifikasi Status Pesanan ke Customer (Real-time)

Skenario: Sistem mengirim notifikasi ke customer ketika status pesanan berubah.

- [] Sistem mengirim notifikasi ketika pesanan berubah status (Menunggu → Diproses → Selesai).
 - [] Notifikasi muncul di halaman customer tanpa harus refresh (websocket/Pusher).
 - [] Notifikasi berisi: nomor order, meja, status terbaru, dan waktu update.
 - [] Jika koneksi customer buruk → sistem menampilkan indikator “menyambungkan kembali”.
 - [] Semua notifikasi tersimpan selama sesi login.
 - [] Customer tidak menerima notifikasi pesanan orang lain.
 - [] Jika customer logout → notifikasi hilang otomatis.
-

AC untuk US027 : Notifikasi ke Kasir Saat Customer Upload Bukti Pembayaran

Skenario: Sistem memberitahukan kasir bahwa customer sudah upload bukti pembayaran.

- [] Ketika customer upload bukti pembayaran → kasir menerima notifikasi real-time.
 - [] Notifikasi berisi: nomor order, nomor meja, dan waktu upload.
 - [] Dashboard kasir menandai order tersebut dengan status “Menunggu Verifikasi Pembayaran”.
 - [] Kasir dapat membuka gambar bukti tanpa harus download.
 - [] Jika upload gagal (format salah, file corrupt) → sistem menolak dan informasikan ke customer.
 - [] Sistem menyimpan riwayat upload untuk audit.
-

AC untuk US028 : Riwayat Pesanan Customer (Hari Ini)

Skenario: Customer melihat riwayat pesanan yang ia buat pada hari yang sama.

- ☐ Customer dapat membuka halaman “Riwayat Pesanan”.
 - ☐ Riwayat hanya menampilkan pesanan milik customer yang sedang login.
 - ☐ Riwayat hanya menampilkan pesanan pada tanggal yang sama (hari ini).
 - ☐ Setiap pesanan menampilkan: nomor order, waktu order, total pembayaran, metode pembayaran, dan status.
 - ☐ Customer dapat membuka detail pesanan untuk melihat item yang dipesan.
 - ☐ Jika belum ada riwayat → tampil pesan “Belum ada pesanan hari ini.”
 - ☐ Riwayat tetap aman meskipun browser di-refresh.
 - ☐ Customer lain tidak bisa melihat riwayat milik user lain (validasi access control).
-

AC untuk US029 : Nota Digital (Customer)

Skenario: Customer melihat dan mengunduh nota digital setelah pesanan selesai.

- ☐ Nota digital tersedia ketika status pesanan = “Selesai”.
- ☐ Nota menampilkan:
 - ☐ Nomor order
 - ☐ Nomor meja
 - ☐ Waktu transaksi
 - ☐ Daftar item + qty + harga
 - ☐ Total harga
 - ☐ Metode pembayaran
- ☐ Nota dapat ditampilkan dalam format digital (HTML/PDF preview).
- ☐ Customer dapat mengunduh nota dalam PDF.

- ☐ QR code unik pada nota (opsional, tetapi recommended untuk validasi).
 - ☐ Nota hanya bisa diakses oleh customer pemilik order tersebut.
 - ☐ Jika customer belum melakukan pembayaran → nota belum tersedia.
-

AC untuk US030 : Laporan Transaksi Harian (Kasir)

Skenario: Kasir melihat laporan pendapatan harian.

- ☐ Kasir dapat memilih tanggal laporan (default = hari ini).
 - ☐ Laporan menampilkan daftar transaksi hari itu lengkap dengan:
 - ☐ No order
 - ☐ Waktu
 - ☐ Total pembayaran
 - ☐ Metode pembayaran (Tunai/QRIS)
 - ☐ Nama kasir (pencatat/verifikator)
 - ☐ Sistem menghitung total pemasukan harian secara otomatis.
 - ☐ Laporan hanya menampilkan transaksi berstatus “Pembayaran Diterima” atau “Selesai”.
 - ☐ Jika tidak ada transaksi → tampil pesan “Belum ada transaksi pada tanggal ini.”
 - ☐ Kasir hanya bisa melihat laporan, tidak bisa mengedit.
 - ☐ Data laporan harus konsisten dengan data order di database (tidak boleh mismatch).
-

AC untuk US031 :Export Laporan Harian ke PDF/Excel

Skenario: Kasir mengekspor laporan transaksi harian ke format PDF atau Excel.

- ☐ Kasir dapat memilih format export: PDF atau Excel.
 - ☐ File export berisi semua transaksi harian lengkap dengan total pemasukan.
 - ☐ Format PDF rapi, terbaca, dan memiliki header perusahaan/UMKM.
 - ☐ Format Excel memiliki kolom yang sesuai (no order, waktu, total, metode pembayaran, kasir).
 - ☐ File berhasil terunduh tanpa corrupt.
 - ☐ Tanggal laporan muncul di bagian header file export.
 - ☐ Data yang diexport harus persis sama dengan data yang ada di halaman laporan.
 - ☐ Jika tidak ada data → export tidak dapat dilakukan dan muncul notifikasi
-

AC untuk US032 : Laporan Mingguan (Kasir)

Skenario: Kasir melihat rekap transaksi dalam satu minggu.

- ☐ Kasir dapat memilih minggu tertentu (minggu berjalan default).
 - ☐ Laporan menampilkan semua transaksi selama 7 hari terakhir.
 - ☐ Sistem menampilkan total pemasukan setiap hari (daily total).
 - ☐ Sistem menampilkan total pemasukan mingguan (weekly total).
 - ☐ Sistem menampilkan jumlah transaksi per hari.
-
- ☐ Kasir dapat membuka detail transaksi dari laporan mingguan.
 - ☐ Jika tidak ada transaksi minggu itu → tampil pesan “Tidak ada transaksi minggu ini.”
 - ☐ Laporan ini hanya untuk view, tidak bisa diedit.

AC untuk US033 : Export Laporan Mingguan ke PDF/Excel

Skenario: Kasir mengekspor laporan mingguan.

- ☐ Kasir dapat memilih format export mingguan (PDF atau Excel).
- ☐ File export menampilkan seluruh transaksi per hari dalam bentuk tabel.
- ☐ Total harian dan total mingguan muncul di bagian bawah laporan.
- ☐ PDF rapi, memiliki header UMKM/cafe/resto.
- ☐ Excel berisi sheet tunggal dengan kolom lengkap.
- ☐ File export tidak corrupt dan sesuai minggu yang dipilih.
- ☐ Jika tidak ada data → export diblok dan muncul notifikasi.

AC untuk US034 : Menu Terlaris (Analitik Sederhana)

Skenario: Kasir melihat daftar menu terlaris berdasarkan jumlah pesanan.

- ☐ Sistem menghitung jumlah item terjual per menu secara otomatis.
- ☐ Kasir dapat memilih rentang waktu (hari ini / minggu ini / custom date).
- ☐ Menu ditampilkanurut berdasarkan yang paling sering dipesan.
- ☐ Setiap item menampilkan: nama menu, jumlah terjual, total pendapatan dari item tersebut.
- ☐ Data update otomatis berdasarkan transaksi yang diverifikasi.
- ☐ Jika tidak ada transaksi → tampilkan pesan “Belum ada data untuk ditampilkan.”
- ☐ Kasir tidak bisa mengedit data, hanya melihat.

Bagian 3 : High Level Design

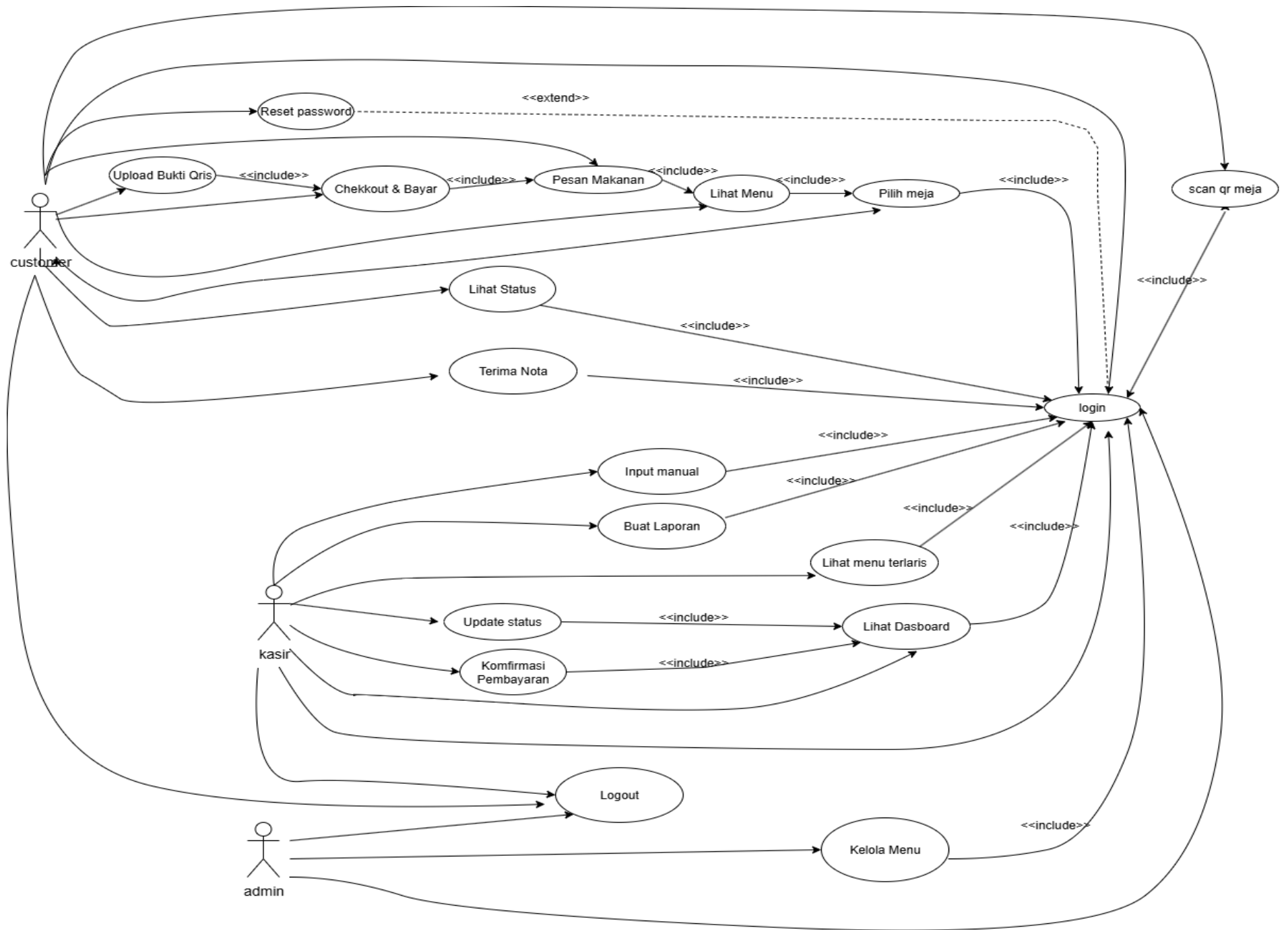
High Level Design (HLD) adalah **desain awal sistem** yang menjelaskan gambaran besar bagaimana sistem akan bekerja tanpa masuk ke detail teknis.

Yang harus dibuat:

1. Use Case Diagram

- Menunjukkan aktor (user/admin) dan fitur-fitur utama yang akan mereka gunakan.
- Contoh: User → Login, Registrasi, Lihat Dashboard; Admin → Kelola Data Produk.
- Gunakan notasi UML standar (oval = use case, stick figure = aktor).

Use Case Diagram merupakan salah satu model dalam Unified Modeling Language (UML) yang berfungsi untuk menggambarkan hubungan antara aktor dengan fungsi-fungsi utama yang terdapat pada sistem. Pada sistem Pool Snack, terdapat tiga aktor utama yang berinteraksi, yaitu Customer, Kasir, dan Admin. Masing-masing aktor memiliki peran dan kebutuhan yang berbeda dalam penggunaan sistem.



1) Aktor dalam Sistem

➤ Customer

Customer merupakan pengguna layanan utama yang melakukan pemesanan makanan dan minuman. Aktivitas yang dapat dilakukan oleh Customer meliputi:

- Login, Logout, dan Reset Password.
- Scan QR Meja dan Pilih Meja untuk menentukan lokasi pemesanan.
- Melihat Menu, Memesan Makanan, melakukan Checkout & Pembayaran, serta mengunggah bukti pembayaran melalui QRIS.
- Melihat Status pesanan yang sedang diproses dan menerima Nota setelah transaksi selesai.

➤ Kasir

Kasir berfungsi untuk mengelola transaksi pembayaran dan pemrosesan pesanan. Aktivitas yang dapat dilakukan Kasir adalah:

- Login dan Logout.
- Melihat Dashboard sebagai pusat informasi.
- Melakukan Konfirmasi Pembayaran, Update Status pesanan, serta Input Manual apabila ada transaksi yang tidak tercatat otomatis.
- Membuat Laporan transaksi dan melihat data Menu Terlaris.

➤ Admin

Admin memiliki peran khusus dalam mengelola konten aplikasi, khususnya menu yang tersedia. Aktivitas Admin antara lain:

- Login dan Logout.
- Mengelola Menu (tambah, ubah, hapus data menu).

2) Hubungan Use Case

Setiap aktor terhubung dengan use case sesuai perannya melalui asosiasi. Beberapa use case juga memiliki hubungan include maupun extend untuk menunjukkan dependensi antar proses:

- Include digunakan untuk menggambarkan bahwa suatu use case harus melibatkan use case lain sebagai prasyarat.

Contoh:

- Lihat Menu meng-include Pilih Meja.
- Pesan Makanan meng-include Lihat Menu.
- Checkout & Bayar meng-include Pesan Makanan.
- Upload Bukti QRIS meng-include Checkout & Bayar.

Dengan demikian, alur aktivitas Customer terbentuk secara berurutan dari login hingga pembayaran selesai.

- Extend digunakan untuk menunjukkan fungsi opsional. Contohnya: Reset Password merupakan ekstensi dari Login, karena hanya dilakukan apabila pengguna gagal masuk.

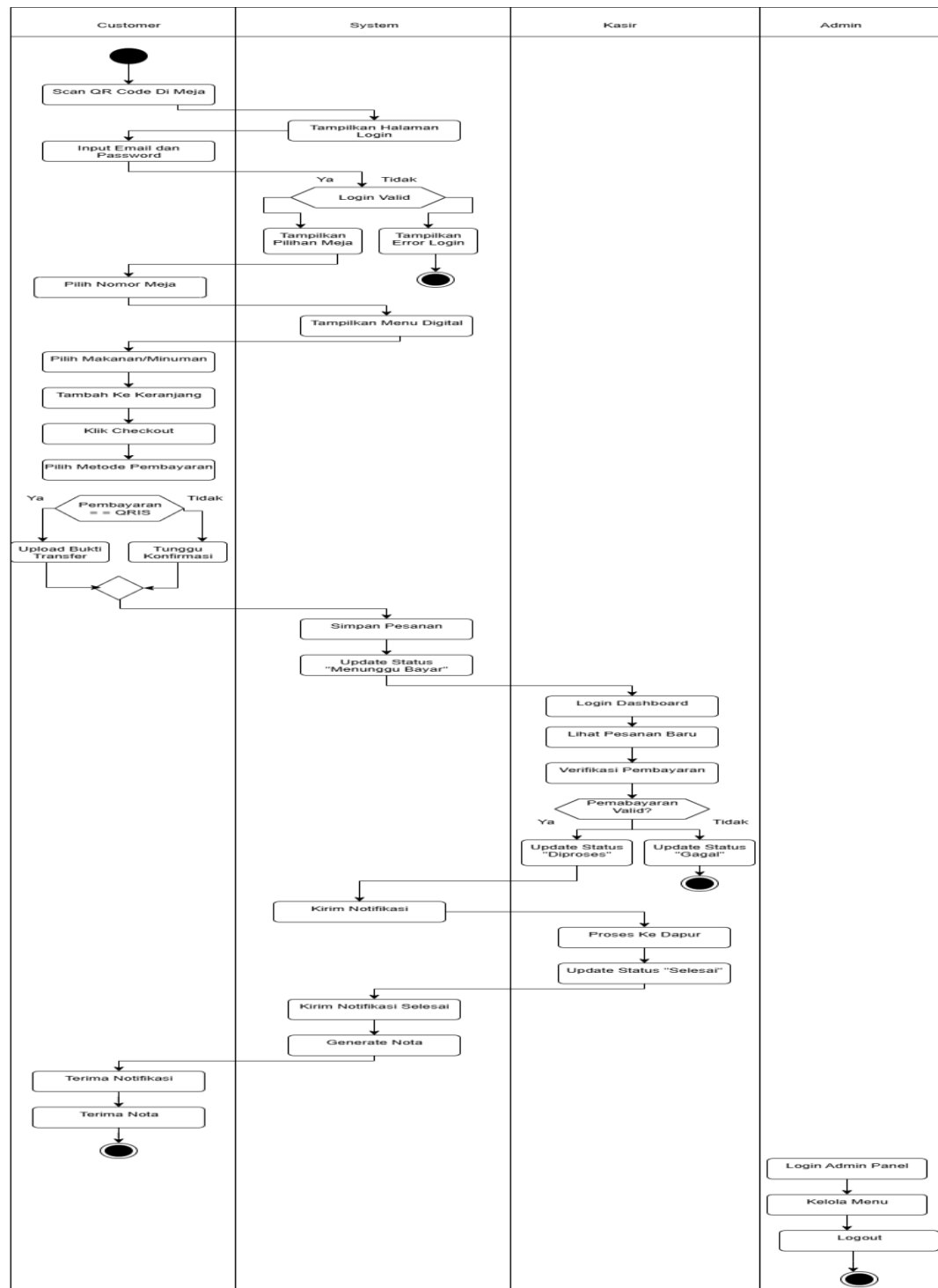
3) Interpretasi Diagram

Berdasarkan diagram yang telah dibuat, sistem *Pool Snack* mampu memfasilitasi proses bisnis inti, yaitu pemesanan makanan/minuman berbasis QR Code. Customer mendapatkan kemudahan dalam melakukan pemesanan dari meja billiard tanpa harus datang langsung ke kasir. Kasir memperoleh dukungan dalam hal verifikasi transaksi serta pembuatan laporan. Sementara itu, Admin berfokus pada pengelolaan data menu agar selalu diperbarui.

Secara keseluruhan, Use Case Diagram ini telah menggambarkan hubungan antaraktor dan fitur utama dalam sistem dengan memanfaatkan notasi standar UML berupa aktor (stick figure), use case (oval), asosiasi (garis), serta relasi include dan extend. Diagram ini juga memperlihatkan alur logis sistem dari awal hingga akhir, sehingga dapat dijadikan acuan dalam tahap perancangan lebih lanjut.

2. Activity Diagram (minimal 1 proses utama)

- Gambarkan alur aktivitas dari salah satu fitur penting (misalnya Login, Input Data, atau Generate Laporan).
- Cukup gunakan simbol dasar (start, decision, action, end).
- Tujuan: agar tim memahami bagaimana proses berjalan dari awal sampai selesai.



Activity Diagram digunakan untuk menggambarkan alur aktivitas (workflow) yang terjadi dalam sistem, termasuk interaksi antaraktor dan sistem itu sendiri. Pada Pool Snack, diagram dibuat dengan pendekatan swimlane, sehingga setiap aktor maupun sistem memiliki “jalur” (lane) masing-masing. Swimlane ini membantu memperjelas tanggung jawab tiap peran dalam proses.

1) Lane Customer

Customer memulai aktivitas dengan memindai QR Code di meja untuk mengakses sistem. Selanjutnya, Customer melakukan login dengan email dan password. Jika login berhasil, sistem akan menampilkan pilihan meja dan menu digital. Customer kemudian:

- Memilih nomor meja.
- Melihat dan memilih makanan/minuman dari menu digital.
- Menambahkan item ke keranjang.
- Melakukan checkout dan memilih metode pembayaran.
- Jika pembayaran dilakukan melalui QRIS, Customer harus mengunggah bukti transfer; jika tidak, sistem menunggu konfirmasi pembayaran dari kasir.
- Setelah pesanan diproses dan pembayaran diverifikasi, Customer akan menerima notifikasi status serta nota transaksi.

2.) Lane System

Sistem bertugas sebagai penghubung yang menampilkan halaman login, memvalidasi kredensial, serta menampilkan menu digital sesuai meja yang dipilih. Sistem juga:

- Menyimpan pesanan.
- Mengubah status pesanan (Menunggu Bayar → Diproses → Selesai).
- Mengirimkan notifikasi status kepada Customer.

- Menghasilkan nota digital setelah transaksi selesai.

3.) Lane Kasir

Kasir berperan dalam memverifikasi transaksi Customer. Setelah login ke dashboard, Kasir dapat:

- Melihat daftar pesanan baru.
- Melakukan verifikasi pembayaran.
 - Jika valid → status pesanan diperbarui menjadi Diproses.
 - Jika tidak valid → status diperbarui menjadi Gagal.
- Kasir kemudian melanjutkan pesanan ke dapur dan mengubah status menjadi Selesai setelah proses selesai.

4.) Lane Admin

Admin memiliki alur aktivitas terpisah dari proses pemesanan. Admin hanya melakukan:

- Login ke panel admin.
- Mengelola data menu (tambah, ubah, hapus).
- Logout setelah selesai.

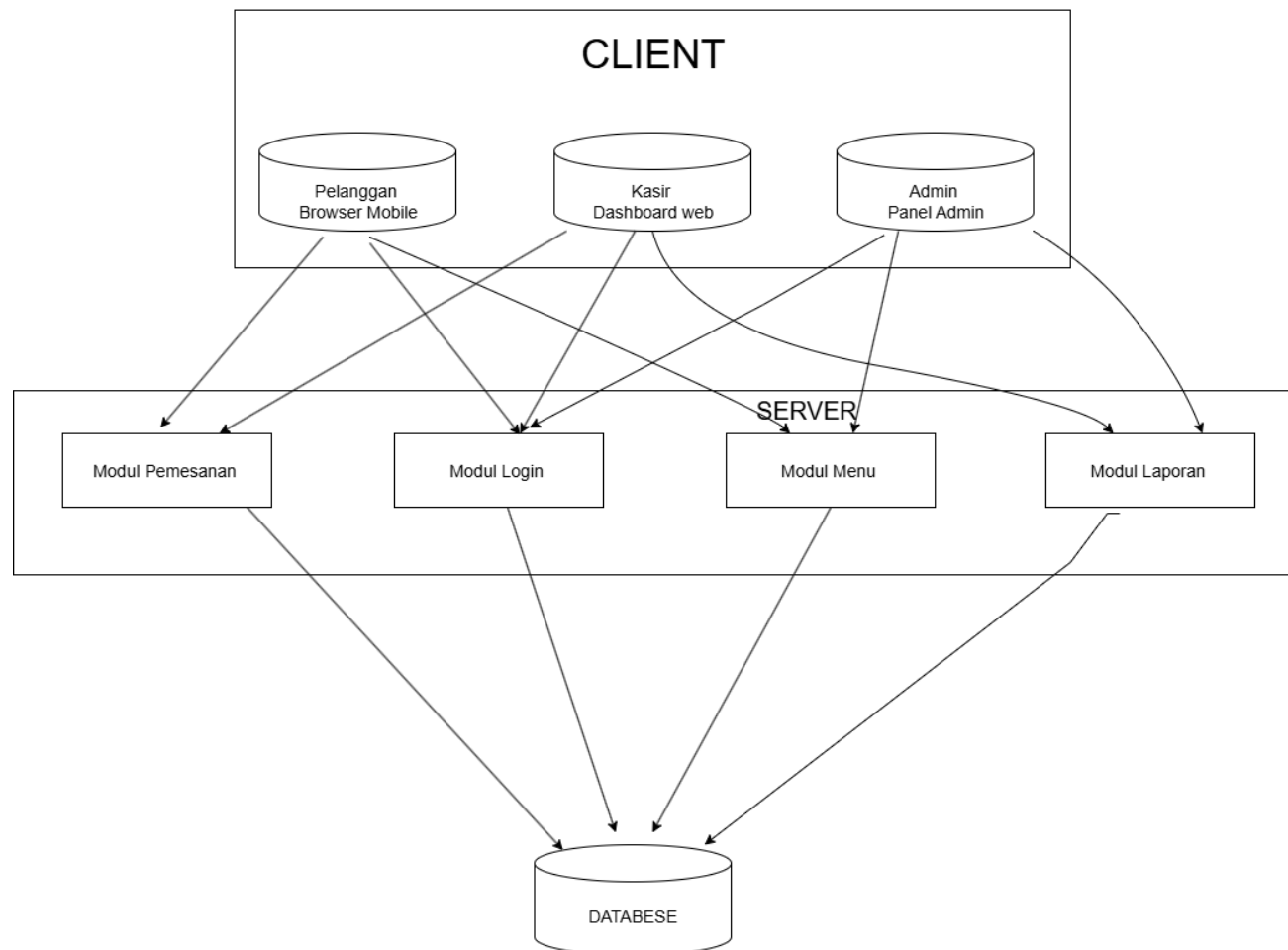
5.) Interpretasi Diagram

Activity Diagram ini menunjukkan integrasi proses bisnis utama dalam sistem Pool Snack. Customer difasilitasi untuk memesan makanan/minuman melalui QR Code, sementara Kasir memastikan pembayaran valid dan pesanan dapat diproses. Sistem berperan sebagai mediator yang mengatur alur data, status, serta notifikasi. Admin mendukung dari sisi pengelolaan menu. Dengan penggunaan swimlane, diagram menjadi jelas dalam menunjukkan siapa melakukan apa, serta bagaimana sistem berinteraksi dengan setiap aktor.

3. Arsitektur Sistem (High Level Architecture)

- Gambaran sederhana bagaimana sistem bekerja secara garis besar.
- Minimal terdiri dari 3 komponen:
 - **Client (User/Browser/App)** → tempat user berinteraksi.
 - **Server/Backend** → logika aplikasi.
 - **Database** → penyimpanan data.
- Bisa juga ditambahkan modul besar (Login, Registrasi, Laporan, Dashboard).
- Bentuknya bisa diagram blok sederhana (tidak perlu detail teknis seperti API atau framework).

Arsitektur Hingh- level Sistem Pool Snack



1) Struktur Diagram

Terdiri dari tiga komponen utama:

- Client: perangkat pengguna (Customer, Kasir, Admin).
- Server: berisi modul-modul inti aplikasi (Login, Pemesanan, Menu, Laporan).
- Database: tempat penyimpanan data terpusat.

2) Client

- Customer menggunakan *browser mobile* untuk mengakses sistem.
- Kasir menggunakan *dashboard web*.
- Admin menggunakan *panel admin*.

3) Server

- Modul Login: autentikasi dan otorisasi untuk semua jenis pengguna.
- Modul Pemesanan: menangani proses order makanan/minuman oleh Customer.
- Modul Menu: menampilkan daftar menu kepada Customer dan memungkinkan Admin melakukan pengelolaan.
- Modul Laporan: menghasilkan laporan transaksi serta data analitik yang dapat diakses oleh Kasir dan Admin.

4) Database

- Menyimpan semua data (user, menu, transaksi, laporan).
- Semua modul di server terkoneksi langsung dengan database.

5) Hubungan

- Customer dapat mengakses modul Login, Pemesanan, dan Menu.
- Kasir mengakses Login, Pemesanan, dan Laporan.
- Admin mengakses Login, Menu, dan Laporan.
- Setiap modul di server melakukan interaksi langsung ke Database sebagai sumber data tunggal.

Penjelasan dari masing – masing kolom tabel user story

1. ID User Story

- Kode unik untuk mengidentifikasi *user story*.
- Contoh: **US-01**, **US-02**, dst.
- Tujuannya agar mudah dilacak & dibahas dalam tim.

2. User Story

- Deskripsi kebutuhan sistem dari sudut pandang pengguna.
- Format standar:
“Sebagai <tipe pengguna>, saya ingin <aksi> agar <manfaat>”.
- Contoh: “Sebagai user, saya ingin login agar dapat mengakses fitur sesuai hak akses saya.”

3. Prioritas

- Tingkat kepentingan/urgensi *user story*.
- Biasanya skala: **Sangat Tinggi, Tinggi, Sedang, Rendah** atau **P1–P4**.
- Tujuannya untuk menentukan fitur mana yang harus dikerjakan dulu.

4. Estimasi (Story Point)

- Ukuran relatif dari **kompleksitas/tingkat usaha** untuk mengerjakan *user story*.
- Tidak sama dengan jam/hari, biasanya pakai skala Fibonacci (1, 2, 3, 5, 8, 13).
- Semakin besar angka → semakin kompleks atau memakan waktu.

5. Status

- Menunjukkan posisi backlog item saat ini.
- Umum digunakan: **To Do, In Progress, Done**.
- Bisa juga ditambah: **Testing, Review**, sesuai workflow tim.

6. Sprint

- Menunjukkan di sprint ke berapa *user story* tersebut akan dikerjakan.
- Contoh: Sprint 1 (minggu pertama), Sprint 2, Sprint 3, dst.
- Membantu mengatur timeline pengerjaan.

Kesimpulan :

- **ID** = identitas.
- **User Story** = kebutuhan pengguna.
- **Prioritas** = tingkat kepentingan.
- **Estimasi** = tingkat kesulitan/kompleksitas. Semakin besar angka semakin menghabiskan waktu
- **Status** = progres pekerjaan.
- **Sprint** = alokasi waktu pengerjaan.

Beberapa catatan penting yang biasanya perlu diperhatikan kelompok Agile (Scrum) saat mengerjakan project RPL:

1. **Fokus pada Iterasi (Sprint)**
 - Pekerjaan dibagi ke dalam sprint (misalnya 1–2 minggu).
 - Setiap sprint harus menghasilkan *increment* yang bisa ditinjau/demonstrasikan.
2. **Product Backlog sebagai Dokumen Utama**
 - Semua kebutuhan, fitur, dan pekerjaan dicatat di product backlog.
 - Prioritas bisa berubah sesuai diskusi dengan *Product Owner*.
3. **User Story Harus Jelas**
 - Menggunakan format standar: *Sebagai [tipe pengguna], saya ingin [aksi] agar [manfaat]*.
 - Lengkapi dengan **Acceptance Criteria** agar jelas kapan dianggap “selesai”.
4. **Estimasi dengan Story Point**
 - Estimasi bukan dalam jam/hari, tapi dalam tingkat kesulitan relatif (1, 2, 3, 5, 8, ...).
 - Berguna untuk menentukan kapasitas tim di tiap sprint.
5. **High Level Design (Opsional tapi Penting)**
 - Meskipun Agile tidak menekankan dokumen tebal, tetap perlu gambaran arsitektur sistem secara umum.
 - Misalnya diagram blok client-server, modul utama, alur data.
6. **Kolaborasi & Komunikasi Intensif**
 - Agile mengutamakan komunikasi langsung, bukan hanya dokumen.
 - Pastikan ada pembagian peran: Scrum Master, Product Owner, Developer.
7. **Review dan Retrospektif**
 - Setelah sprint selesai, lakukan *sprint review* (demo hasil kerja).
 - Lanjutkan dengan *retrospektif* untuk evaluasi cara kerja tim.