

## A. Checklist Pembelajaran Mandiri: Analisis & Perancangan Sistem

### 1. Pemahaman Domain Sistem OBE (Kurikulum Berbasis Capaian)

#### 1.1 Pahami Konsep Outcome-Based Education (OBE), PPEPP, dan Struktur Kurikulum OBE (PL–CPL–BK–MK–CPMK–RPS)

Level	Fokus Pembelajaran	Deskripsi
Level 1	Mengenal istilah OBE & CPL	Membaca definisi OBE, CPL, dan perbedaan dengan kurikulum konvensional.
Level 2	Struktur dasar OBE	Memahami hubungan antar elemen PL–CPL–BK–MK secara konseptual (tanpa pemetaan teknis).
Level 3	Pemetaan antar elemen	Belajar bagaimana CPL diturunkan dari PL, dan bagaimana MK & CPMK mendukung CPL.
Level 4	Integrasi RPS dalam OBE	Memahami bagaimana CPMK diturunkan dalam RPS mingguan dan dikaitkan dengan penilaian.
Level 5	Skema traceability OBE	Mampu menjelaskan jejak capaian: PL → CPL → CPMK → Assessment → Nilai → CPL kembali. Dihubungkan dengan penilaian berbasis bukti.

#### 1.2 Pelajari Siklus PPEPP (Perencanaan–Pelaksanaan–Evaluasi–Pengendalian–Peningkatan)

Level	Fokus Pembelajaran	Deskripsi
Level 1	Kenali siklus PPEPP	Mengenali 5 tahap siklus PPEPP dan urutannya.
Level 2	Contoh praktik PPEPP	Pelajari contoh PPEPP dari Prodi SI (mis. peninjauan kurikulum).
Level 3	Kaitan PPEPP dengan OBE	Menjelaskan bagaimana PPEPP mendukung implementasi OBE di prodi.
Level 4	PPEPP dalam sistem informasi	Identifikasi titik dalam sistem (mis. dashboard CPL, log refleksi) yang mencerminkan PPEPP.
Level 5	Desain fitur PPEPP digital	Mendesain fitur yang mampu mencatat siklus PPEPP, misalnya dashboard siklus mutu atau log CQI.

#### 1.3 Tinjau Dokumen BRD, User Flow, ERD, dan Modul-Modul Utama

Level	Fokus Pembelajaran	Deskripsi
Level 1	Pahami struktur dokumen BRD	Membaca isi utama BRD: Tujuan, Fitur, Stakeholder, Modul.
Level 2	Identifikasi modul utama	Mengenali modul penting: CPL, CPMK, RPS, Penilaian, Dashboard.
Level 3	Telaah user flow & ERD	Memahami bagaimana entitas (PL–CPL–BK–MK) saling terkait dalam ERD dan user flow.

<b>Level 4</b>	Koneksikan ke kebutuhan pengguna	Menjelaskan bagaimana tiap modul menjawab kebutuhan user (kaprodi, dosen, mahasiswa).
<b>Level 5</b>	Breakdown modul ke rencana dev	Mampu menerjemahkan deskripsi modul menjadi struktur API & tampilan frontend yang akan dikembangkan.

## 2. Business Process Mapping

### 2.1 Buat/validasi BPMN diagram untuk modul utama (RPS, CPMK, Penilaian, TA, Refleksi)

<b>Level</b>	<b>Fokus Pembelajaran</b>	<b>Deskripsi</b>
<b>Level 1</b>	Kenali elemen dasar BPMN	Pelajari simbol BPMN (start, task, decision, swimlane, event) dan fungsinya.
<b>Level 2</b>	Latihan BPMN sederhana	Buat flow proses dasar (mis. “Input CPMK” oleh dosen) dalam 4–5 langkah.
<b>Level 3</b>	BPMN satu modul	Buat proses BPMN lengkap untuk 1 modul (mis. RPS: input → validasi → publish).
<b>Level 4</b>	BPMN lintas peran	Buat BPMN yang melibatkan 2–3 peran (e.g., dosen, kaprodi, mahasiswa – proses TA).
<b>Level 5</b>	Review & validasi proses real	Bandingkan flow BPMN dengan proses nyata prodi, perbaiki inkonsistensi, dan siapkan sebagai acuan implementasi sistem.

### 2.2 Identifikasi aktor, input, output, dan rule di tiap modul (sesuai BRD)

<b>Level</b>	<b>Fokus Pembelajaran</b>	<b>Deskripsi</b>
<b>Level 1</b>	Kenali peran (aktor) utama	Kenali aktor: Kaprodi, Dosen, Admin, Mahasiswa, Tim Akreditasi.
<b>Level 2</b>	Tentukan input & output	Pelajari contoh input/output tiap modul, misalnya input RPS (CPMK, Materi) → Output (dokumen RPS).
<b>Level 3</b>	Pahami rule di tiap modul	Identifikasi aturan utama, seperti: “Setiap MK minimal memiliki 1 CPMK” atau “RPS harus disetujui Kaprodi sebelum aktif.”
<b>Level 4</b>	Modelkan relasi antar rule	Gabungkan input-output-rule ke dalam flow operasional per modul.
<b>Level 5</b>	Validasi & susun mapping tabular	Susun dokumen final yang memetakan: Modul – Input – Output – Role – Rule. (dapat digunakan tim dev dan QA).

### 2.3 Buat flow data antar entitas (misal MK ↔ CPMK ↔ CPL)

<b>Level</b>	<b>Fokus Pembelajaran</b>	<b>Deskripsi</b>
<b>Level 1</b>	Kenali relasi antar entitas	Pelajari relasi dasar di ERD: 1 MK → N CPMK, M CPMK ↔ N CPL.
<b>Level 2</b>	Buat diagram data flow sederhana	Gambar relasi data CPL → CPMK → Nilai CPMK.

<b>Level 3</b>	Visualisasi dependensi antar modul	Buat diagram alir data antar modul (mis. CPMK ↔ Penilaian ↔ Dashboard CPL).
<b>Level 4</b>	Sertakan transformasi data	Identifikasi bagaimana data berubah antar modul, misalnya: “nilai CPMK + bobot MK → nilai CPL”.
<b>Level 5</b>	Uji alur & sinkronisasi data	Lakukan walkthrough dari input awal (CPMK) → hasil akhir (Dashboard CPL mahasiswa) dan pastikan tidak ada missing-link.

### 3. Data Structure & ERD Review

#### 3.1 Pahami dan review ulang ERD sistem (entitas utama, relasi, PK/FK)

<b>Level</b>	<b>Fokus Pembelajaran</b>	<b>Deskripsi</b>
<b>Level 1</b>	Kenali konsep entitas dan relasi	Pahami perbedaan entitas, atribut, dan relasi (1:N, M:N, 1:1).
<b>Level 2</b>	Telusuri entitas utama OBE	Identifikasi entitas penting: PL, CPL, BK, MK, CPMK, RPS, Mahasiswa, Nilai.
<b>Level 3</b>	Pelajari kunci (PK/FK) & normalisasi	Pelajari bagaimana PK/FK menghubungkan tabel, serta bentuk normalisasi sederhana (1NF–3NF).
<b>Level 4</b>	Telusuri rantai relasi OBE	Visualisasikan jejak PL → CPL → CPMK → Penilaian → Nilai CPL di ERD.
<b>Level 5</b>	Identifikasi potensi duplikasi & optimasi struktur	Uji ERD: Apakah bisa menangani multi versi kurikulum? Apakah relasi M:N perlu tabel junction? Cek indexing.

#### 3.2 Latihan membuat skema tabel dalam PostgreSQL berdasarkan kurikulum OBE

<b>Level</b>	<b>Fokus Pembelajaran</b>	<b>Deskripsi</b>
<b>Level 1</b>	Buat tabel sederhana	Latihan membuat 1 tabel (misalnya: mata_kuliah) dengan tipe data dasar (varchar, int).
<b>Level 2</b>	Tambah constraint dasar	Tambahkan PK, NOT NULL, UNIQUE constraint.
<b>Level 3</b>	Buat relasi antar tabel	Tambahkan FK antara tabel (misalnya: cpmk → mk, mk → kurikulum).
<b>Level 4</b>	Tambah constraint lanjutan	Coba gunakan CHECK constraint, ENUM, dan tipe data ARRAY/JSONB.
<b>Level 5</b>	Simulasikan query lintas tabel	Buat query dengan JOIN antar 3–4 tabel (mis. mahasiswa → enrollment → nilai_cpmk → cpl).

### 3.3 Identifikasi field yang wajib di-index, tabel yang perlu partisi (nilai, audit log)

Level	Fokus Pembelajaran	Deskripsi
Level 1	Pahami konsep index	Pelajari tujuan index: mempercepat pencarian/filter data besar.
Level 2	Identifikasi FK & field unik	Tandai semua kolom yang sering di-join atau dicari sebagai kandidat index (mis. kode_cpl, mk_id, mahasiswa_id).
Level 3	Buat indeks dasar di PostgreSQL	Latihan perintah CREATE INDEX dan uji performa dengan EXPLAIN.
Level 4	Pahami partisi tabel besar	Pelajari PARTITION BY untuk tabel besar (mis. nilai_cpmk per tahun).
Level 5	Rancang strategi indexing & partisi	Buat daftar indeks yang diperlukan per modul + strategi partisi (audit_log by tanggal, nilai by semester).

## 4. API Design Planning

### 4.1 Pelajari prinsip RESTful API design (GET/POST/PUT/DELETE)

Level	Fokus Pembelajaran	Deskripsi
Level 1	Kenali metode HTTP dasar	Pahami fungsi GET, POST, PUT, DELETE dalam konteks API.
Level 2	Coba API publik	Gunakan Postman untuk uji coba API publik (misal: JSONPlaceholder).
Level 3	Pelajari RESTful endpoint pattern	Pelajari konvensi URI /cpl, /mk/{id}, /rps/{id}/weeks dan nested resource.
Level 4	Tentukan status code dan response standar	Pelajari HTTP status codes (200 OK, 201 Created, 400 Bad Request, dll) dan format JSON standar.
Level 5	Buat dokumentasi mini API	Susun dokumentasi 5–10 endpoint awal menggunakan Notion, Swagger YAML, atau ReDoc.

### 4.2 Buat rencana endpoint utama (dalam Swagger/OpenAPI YAML if possible)

Level	Fokus Pembelajaran	Deskripsi
Level 1	Buat daftar resource	Daftar modul utama: CPL, CPMK, MK, RPS, Mahasiswa, Penilaian.
Level 2	Tentukan endpoint dasar per resource	Misalnya: GET /mk, POST /cpl, PUT /cpl/{id}
Level 3	Susun struktur endpoint kompleks	Rancang endpoint nested: POST /rps/{id}/weeks, GET /cpl/{id}/contributions
Level 4	Gunakan Swagger Editor	Tulis file .yaml menggunakan Swagger untuk mendefinisikan endpoint, parameter, dan response
Level 5	Validasi & share API spec	Review bersama tim, tes rendering dokumentasi, pastikan konsisten dan dapat dimengerti oleh frontend developer.

## 4.3 Rancang JSON schema untuk payload & response (e.g., CPL response, RPS payload)

Level	Fokus Pembelajaran	Deskripsi
Level 1	Kenali format JSON & key-value	Pelajari bentuk dasar { "kode_cpl": "CPL1", "deskripsi": "..." }
Level 2	Buat JSON sederhana untuk 1 entitas	Latihan JSON untuk mata_kuliah, profil_lulusan, dll
Level 3	Buat nested JSON	Contoh: CPMK yang memuat array contribution_to_cpl, rps yang memuat weeks []
Level 4	Gunakan JSON schema validator	Latihan validasi struktur JSON di <a href="https://jsonlint.com">https://jsonlint.com</a> atau tools sejenis
Level 5	Buat kontrak frontend-backend	Susun JSON untuk form input + response per modul agar dapat diintegrasikan oleh frontend (React) dan backend (Go).

## 4.4 Pahami pola versioning, rate limit, dan auth token API

Level	Fokus Pembelajaran	Deskripsi
Level 1	Kenali versioning API	Pahami URI versioning (/api/v1/, /api/v2/)
Level 2	Pelajari auth token dasar	Pelajari skema Bearer token, penggunaan Authorization Header
Level 3	Implementasi JWT token dasar	Gunakan JWT di backend Go (misal dengan <a href="https://github.com/golang-jwt/jwt/v4">github.com/golang-jwt/jwt/v4</a> )
Level 4	Pahami rate limit API	Pahami konsep limit x request per menit, gunakan middleware (mis. Gin limiter)
Level 5	Simulasikan client-token-API	Buat flow login → dapat token → akses endpoint terproteksi dengan Postman / React.

## 5. Modul Khusus untuk Disiapkan Desain Awal

### 5.1 Modul Kurikulum & Mapping CPL–CPMK

Level	Fokus Pembelajaran	Deskripsi
Level 1	Kenali struktur data CPL dan CPMK	Pahami elemen kode_cpl, kategori, kode_cpmk, bobot, dan relasinya.
Level 2	Buat form input data CPL & CPMK	Latihan desain form sederhana (frontend/JSON payload).
Level 3	Buat tabel pemetaan CPL–CPMK	Desain UI untuk checkbox matriks CPL × CPMK.
Level 4	Simulasikan penyimpanan relasi	Buat tabel junction di PostgreSQL: cpmk_cpl_map(cpmk_id, cpl_id, weight).
Level 5	Desain validasi & kalkulasi otomatis	Pastikan total bobot = 100%; sistem memberi warning jika mapping tidak lengkap.

## 5.2 Modul RPS Mingguan

Level	Fokus Pembelajaran	Deskripsi
Level 1	Kenali elemen RPS	Pahami: sub-CPMK, minggu, metode, kriteria, bobot.
Level 2	Buat struktur tabel RPS per minggu	Desain form per minggu dan struktur data array untuk RPS.
Level 3	Desain repeatable input UI (React)	Gunakan map/loop untuk input 16 minggu pembelajaran.
Level 4	Tambahkan validasi bobot penilaian	Validasi jumlah bobot 100% otomatis pada form.
Level 5	Desain export RPS ke PDF/Word	Integrasi template ekspor yang sesuai dengan institusi.

## 5.3 Modul Penilaian & Capaian CPL

Level	Fokus Pembelajaran	Deskripsi
Level 1	Pahami konsep nilai CPMK & CPL	Pelajari rumus nilai CPL = agregat nilai CPMK × bobot.
Level 2	Buat form input nilai komponen	Input nilai per mahasiswa per komponen: UTS, tugas, quiz.
Level 3	Kalkulasi nilai CPMK per mahasiswa	Gunakan formula: $\Sigma (\text{nilai} \times \text{bobot}) / \text{total bobot}$ .
Level 4	Agregasi ke nilai CPL	Gunakan tabel relasi MK/CPMK → CPL untuk menghitung nilai CPL.
Level 5	Visualisasi dashboard CPL	Desain radar chart per mahasiswa dan heatmap agregat prodi.

## 5.4 Modul Tugas Akhir (Logbook, Desk Eval, Sidang)

Level	Fokus Pembelajaran	Deskripsi
Level 1	Kenali alur TA1–TA2	Pahami tahapan: pengajuan → bimbingan → seminar → sidang.
Level 2	Buat form logbook bimbingan	Input waktu, ringkasan diskusi, status revisi.
Level 3	Desain evaluasi proposal	Input rubrik evaluasi proposal dan hasil review.
Level 4	Buat form penilaian TA final	Rubrik nilai akhir TA: presentasi, isi, teknis, CPL terkait.
Level 5	Hubungkan TA ke CPL	Simulasikan penilaian TA berkontribusi ke CPL tertentu.

## 5.5 Modul Dashboard PPEPP

Level	Fokus Pembelajaran	Deskripsi
Level 1	Kenali indikator PPEPP	Misalnya: % MK dengan RPS lengkap, % CPL tercapai.
Level 2	Buat layout dashboard awal	Gunakan cards, chart, dan warna untuk indikator.
Level 3	Implementasi filter data	Filter berdasarkan semester, prodi, dosen.
Level 4	Integrasi data kalkulasi CPL/CPMK	Gunakan hasil hitungan real-time dari backend.
Level 5	Tambahkan CQI log (perbaikan berkelanjutan)	Fitur pelaporan evaluasi per dosen + tindak lanjut per MK.

## B. Checklist Pembelajaran Mandiri: Development System

### 6. React.js (Frontend)

#### 6.1 Review dasar React: props, state, hooks (`useState`, `useEffect`, `useContext`)

Level	Fokus Pembelajaran	Deskripsi
Level 1	Struktur dasar komponen	Buat komponen fungsi sederhana, pahami JSX dan return.
Level 2	Props dan state lokal	Latihan passing props dan <code>useState</code> untuk menangani input.
Level 3	Side effect dengan <code>useEffect</code>	Ambil data dari dummy API dan render ke tampilan.
Level 4	Global state dengan <code>useContext</code>	Simulasikan data pengguna login disimpan di context global.
Level 5	Strukturisasi modul & komponen reusable	Bagi halaman menjadi komponen modular (FormCPL, TabelCPMK, SidebarNavigation).

#### 6.2 Pelajari `react-router` untuk navigasi antar modul

Level	Fokus Pembelajaran	Deskripsi
Level 1	Setup routing sederhana	Gunakan <code>react-router-dom</code> untuk membuat dua halaman ( <code>/home</code> , <code>/cpl</code> ).
Level 2	Route dengan parameter	Buat halaman detail CPL dengan route <code>/cpl/:id</code> .
Level 3	Navigasi & link aktif	Tambahkan sidebar dan highlight menu aktif.
Level 4	Protected route	Buat route yang hanya bisa diakses saat login (menggunakan context token).
Level 5	Nested routing & layout	Buat route bertingkat untuk modul RPS ( <code>/rps/:id/weeks/:weekNo</code> ) dengan layout parent.

### 6.3 Buat latihan komponen modular (tabel RPS mingguan, form mapping CPL–CPMK)

Level	Fokus Pembelajaran	Deskripsi
<b>Level 1</b>	Buat form input sederhana	Form input untuk <code>kode_cpl</code> , <code>deskripsi</code> , <code>kategori</code> .
<b>Level 2</b>	Buat tabel statis	Tampilkan array data dummy dalam bentuk tabel.
<b>Level 3</b>	Tambahkan fitur dinamis	Tambah kemampuan add/remove baris untuk RPS mingguan.
<b>Level 4</b>	Integrasi dengan state	Tabel update otomatis saat input berubah ( <code>useState</code> ).
<b>Level 5</b>	Komponen reusable & prop-driven	Buat <code>FormBuilder</code> dan <code>TableEditor</code> yang bisa dipakai ulang lintas modul.

### 6.4 Pelajari form validation (React Hook Form, Yup)

Level	Fokus Pembelajaran	Deskripsi
<b>Level 1</b>	Form input dasar dengan <code>react-hook-form</code>	Implementasi form sederhana dengan validasi <code>required</code> .
<b>Level 2</b>	Validasi lanjutan dengan <code>Yup</code>	Validasi panjang teks, angka minimum, format email.
<b>Level 3</b>	Validasi kompleks per aturan OBE	Validasi: total bobot = 100%, CPMK harus dipetakan.
<b>Level 4</b>	Error message & highlighting	Tampilkan pesan error di bawah input yang salah.
<b>Level 5</b>	Validasi dinamis	Validasi array (mis. 16 minggu RPS) dengan nested schema.

### 6.5 Pelajari integrasi REST API (`axios/fetch`) dan state management (`useContext / Redux`)

Level	Fokus Pembelajaran	Deskripsi
<b>Level 1</b>	Fetch data dari API dummy	Gunakan <code>axios.get()</code> ke API publik dan tampilkan data.
<b>Level 2</b>	Form submit ke backend	Kirim data <code>POST</code> ke endpoint mock dan tangani respons.
<b>Level 3</b>	Global auth state dengan <code>useContext</code>	Simpan token login dan inject ke header API.
<b>Level 4</b>	Integrasi loading, error, dan feedback	Tambahkan indikator loading & pesan sukses/gagal.
<b>Level 5</b>	Gunakan Redux Toolkit (opsional)	Implementasi skala besar: store modular per modul (mis. store untuk RPS, CPMK).

## 6.6 Pelajari penggunaan chart library (e.g., Chart.js, Recharts) untuk visualisasi CPL

Level	Fokus Pembelajaran	Deskripsi
Level 1	Kenali jenis chart yang umum	Pahami kapan menggunakan bar, line, pie, radar chart.
Level 2	Tampilkan data statis dalam chart	Buat bar chart pencapaian CPL per mahasiswa dari array dummy.
Level 3	Gunakan Recharts untuk radar chart CPL	Visualisasikan 6–10 CPL per mahasiswa dalam format spider chart.
Level 4	Integrasi chart dengan API	Ambil data nilai CPL dari backend dan render secara dinamis.
Level 5	Tambah interaktivitas & filter	Tambahkan tooltip, highlight CPL rendah, dan filter semester.

## 7. Golang (Backend)

### 7.1 Review dasar Golang syntax, struct, interface, error handling

Level	Fokus Pembelajaran	Deskripsi
Level 1	Sintaks dasar Go	Pahami package main, func, import, dan tipe data dasar ( <code>int, string, bool</code> ).
Level 2	Struct dan slice	Buat struct ProfilLulusan, CPL dan gunakan dalam array/slice.
Level 3	Error handling dasar	Tangani error menggunakan <code>if err != nil</code> dan <code>log.Fatal</code> .
Level 4	Interface dan fungsi modular	Buat interface Repo dan implementasi RepoPostgres untuk testing.
Level 5	Unit test sederhana	Buat file <code>_test.go</code> untuk menguji fungsi manipulasi data (tanpa DB dulu).

### 7.2 Setup router & handler (e.g., gin-gonic, echo)

Level	Fokus Pembelajaran	Deskripsi
Level 1	Instalasi Gin/Echo	Setup proyek menggunakan Go modules dan install Gin/Echo.
Level 2	Definisikan endpoint dasar	Buat GET /hello dan POST /test untuk testing.
Level 3	Struktur handler	Pisahkan routes.go, handlers/, dan models/.
Level 4	Gunakan parameter dan query	Implementasi GET /cpl/:id dan GET /cpl?keyword=x.
Level 5	Setup routing modular per fitur	Buat grup route api/v1/pl, api/v1/mk, api/v1/rps masing-masing dalam file sendiri.

### 7.3 Buat latihan REST API (CRUD Profil Lulusan, CPL, MK)

Level	Fokus Pembelajaran	Deskripsi
Level 1	Buat POST & GET	Implementasi POST /pl dan GET /pl/:id.
Level 2	Buat PUT & DELETE	Implementasi edit dan hapus data.
Level 3	Simpan ke memory/map	Simulasi penyimpanan data sementara menggunakan map atau slice.
Level 4	Integrasi dengan PostgreSQL (tanpa ORM)	Koneksi langsung pakai database/sql untuk simpan ambil data CPL.
Level 5	Refactor dengan repository pattern	Pisahkan controller, service, dan repo agar scalable dan testable.

### 7.4 Pelajari middleware auth/JWT dan error middleware

Level	Fokus Pembelajaran	Deskripsi
Level 1	Pelajari JWT dasar	Pahami isi JWT (header, payload, signature).
Level 2	Generate token login	Implementasikan POST /login → kembalikan JWT token.
Level 3	Middleware validasi JWT	Cek token di Authorization header untuk akses endpoint tertentu.
Level 4	Tambah peran pengguna	Tambahkan role: admin, kaprodi, dosen, mahasiswa dalam token dan validasi akses.
Level 5	Global error handler	Tangani error 401, 403, dan response JSON standar di middleware.

### 7.5 Implementasi data binding dan validator struct

Level	Fokus Pembelajaran	Deskripsi
Level 1	Binding struct	Gunakan c.Bind(&req) di Gin untuk POST data JSON ke struct.
Level 2	Validasi required field	Gunakan tag binding:"required" pada field struct.
Level 3	Validasi nilai (min/max/email)	Tambahkan binding:"gte=1,lte=5" atau validasi email.
Level 4	Kustomisasi pesan error	Buat middleware atau response formatter untuk tampilkan error friendly.
Level 5	Validasi nested struct & array	Buat validasi pada array []CPMK dan nested struct di payload RPS.

## 8. PostgreSQL (Database)

### 8.1 Review DDL & DML (CREATE TABLE, JOIN, WHERE, GROUP BY)

Level	Fokus Pembelajaran	Deskripsi
Level 1	Buat tabel dasar	Tulis CREATE TABLE untuk entitas sederhana seperti cpl, mk, mahasiswa.
Level 2	Masukkan dan baca data	Latihan INSERT, SELECT, dan UPDATE data.

<b>Level 3</b>	Gunakan filter dan sort	Terapkan WHERE, ORDER BY, LIMIT, OFFSET.
<b>Level 4</b>	Gabungkan data dengan JOIN	Buat JOIN antara mk dan cpl melalui pemetaan mk_cpl_map.
<b>Level 5</b>	Gunakan agregasi	Terapkan GROUP BY, COUNT, AVG untuk menghitung jumlah MK per CPL atau nilai rata-rata CPMK.

## 8.2 Latihan query kompleks (menghitung nilai CPL dari nilai CPMK)

<b>Level</b>	<b>Fokus Pembelajaran</b>	<b>Deskripsi</b>
<b>Level 1</b>	Pahami struktur nilai CPMK	Pelajari tabel nilai_cpmk, cpmk_cpl_map, mk.
<b>Level 2</b>	Hitung nilai CPMK rata-rata	Query nilai CPMK mahasiswa untuk satu MK.
<b>Level 3</b>	Gabungkan CPMK ke CPL	Gunakan JOIN antar CPMK → CPL dan hitung rata-rata kontribusi.
<b>Level 4</b>	Terapkan bobot kontribusi	Kalikan nilai CPMK × bobot → sum → bagi total bobot.
<b>Level 5</b>	Buat view atau query final CPL	Buat query agregat CPL per mahasiswa per semester sebagai sumber dashboard.

## 8.3 Pelajari indexing, constraint, dan materialized view

<b>Level</b>	<b>Fokus Pembelajaran</b>	<b>Deskripsi</b>
<b>Level 1</b>	Buat primary key & foreign key	Tambahkan PK/FK pada mk, cpmk, cpl, dan pemetaan antar entitas.
<b>Level 2</b>	Gunakan UNIQUE & CHECK	Validasi kode_cpl unik per kurikulum, sks > 0, kategori dalam enum.
<b>Level 3</b>	Index untuk pencarian cepat	Gunakan CREATE INDEX di kode_mk, nim, periode.
<b>Level 4</b>	Buat materialized view	Simulasikan view nilai_cpl_per_mahasiswa untuk cache data dashboard.
<b>Level 5</b>	Refresh dan optimasi performa	Buat scheduler untuk REFRESH MATERIALIZED VIEW, ukur waktu eksekusi.

## 8.4 Pelajari JSONB dan array (untuk kolom referensi fleksibel)

<b>Level</b>	<b>Fokus Pembelajaran</b>	<b>Deskripsi</b>
<b>Level 1</b>	Buat kolom JSONB	Tambahkan kolom referensi_standar di bk atau rps.
<b>Level 2</b>	Simpan array referensi	Masukkan JSON array: ["KKNI", "CC2020"] .
<b>Level 3</b>	Query isi JSON	Gunakan ->, ->>, dan @> untuk filter WHERE referensi @> '["CC2020"]' .
<b>Level 4</b>	Index JSONB	Gunakan GIN index untuk mempercepat pencarian dalam JSONB.
<b>Level 5</b>	Integrasi API ↔ JSONB	Gunakan Go untuk parse JSON input dan simpan langsung ke kolom PostgreSQL JSONB.

## 9. API Integration

### 9.1 Buat skema integrasi React ↔ Go API

Level	Fokus Pembelajaran	Deskripsi
Level 1	Pahami alur client-server	Pelajari alur data: input form di React → dikirim ke API Go → disimpan ke DB.
Level 2	Uji GET API dari React	Gunakan <code>axios.get()</code> untuk mengambil data dari endpoint Go (misal: <code>/api/cpl</code> ).
Level 3	Uji POST API dari form React	Submit form (misal form MK) menggunakan <code>axios.post()</code> dan tampilkan respons.
Level 4	Tambah loading dan error handler	Tampilkan spinner saat menunggu respons, dan pesan error jika gagal.
Level 5	Strukturkan folder integrasi	Buat folder <code>/services/</code> atau <code>/api/</code> di React untuk centralisasi pemanggilan API.

### 9.2 Latihan pengiriman form dari React ke Go API (via axios)

Level	Fokus Pembelajaran	Deskripsi
Level 1	Buat form input sederhana	Form input CPL di React dengan validasi minimal.
Level 2	Submit ke Go API lokal	Kirim data CPL ke backend menggunakan <code>axios.post()</code> ke endpoint lokal ( <code>localhost:8080</code> ).
Level 3	Tampilkan notifikasi hasil	Tambahkan notifikasi sukses/gagal setelah submit.
Level 4	Reset form otomatis	Kosongkan form dan redirect ke list setelah berhasil submit.
Level 5	Integrasi form dinamis (e.g. matrix CPMK-CPL)	Kirim data array dari mapping form (checkbox matrix) ke backend dan simpan di junction table.

### 9.3 Implementasi CORS handling dan API security (token/jwt)

Level	Fokus Pembelajaran	Deskripsi
Level 1	Kenali masalah CORS	Pelajari mengapa <code>Access-Control-Allow-Origin</code> dibutuhkan saat React dan Go berjalan di port berbeda.
Level 2	Tambahkan CORS middleware	Konfigurasi Gin atau Echo untuk menerima request dari React ( <code>localhost:3000</code> ).
Level 3	Implementasi token JWT	Kirim token di header: <code>Authorization: Bearer &lt;token&gt;</code> dari React ke backend.
Level 4	Validasi token di backend	Tambahkan middleware auth di Go untuk memfilter akses endpoint.
Level 5	Expired token handling	Tangani kondisi token expired dengan logout otomatis dan redirect ke halaman login.

## 9.4 Logging & error handler di backend dan frontend

Level	Fokus Pembelajaran	Deskripsi
Level 1	Logging dasar di Go	Gunakan <code>log.Println()</code> atau <code>logrus</code> untuk mencatat permintaan dan error.
Level 2	Logging respons API	Tambahkan log untuk setiap endpoint: method, path, user, status.
Level 3	Tangani error dengan kode	Gunakan struktur respons standar: <code>{ success: false, message: "", errors: [] }</code> .
Level 4	Tampilkan error di React	Tampilkan alert atau snackbar jika API gagal diakses atau validasi gagal.
Level 5	Logging terpusat	(Opsional) Kirim log ke file atau sistem log eksternal seperti ELK atau Grafana Loki.