**Dokumen Desain Perangkat Lunak (SDD) Untuk Modul Registrasi Mata Kuliah (KRS)**

**1. Pendahuluan**

**1.1 Tujuan**

Dokumen ini bertujuan untuk menjelaskan desain arsitektur, komponen, dan implementasi teknis dari Modul Registrasi Mata Kuliah (KRS) berbasis web. Dokumen ini menjadi panduan bagi tim pengembang, QA, dan stakeholder dalam memastikan sistem memenuhi kebutuhan fungsional dan non-fungsional, seperti keamanan, skalabilitas, dan integrasi dengan sistem akademik lainnya.

**1.2 Ruang Lingkup**

Modul KRS dirancang untuk menyediakan fitur-fitur berikut:

1. **Autentikasi dan otorisasi pengguna** berbasis peran (admin, mahasiswa, dosen).
2. **Registrasi mata kuliah** dengan validasi prasyarat, kapasitas kelas, dan deteksi konflik jadwal.
3. **Dashboard interaktif** untuk melihat status registrasi, jadwal kuliah, dan transkrip sementara.
4. **Integrasi API** dengan sistem nilai, pembayaran, dan jadwal akademik.
5. **Pembuatan laporan** statistik registrasi untuk keputusan administrasi.

**1.3 Referensi**

1. **IEEE 1016-2009:** Standar deskripsi desain perangkat lunak.
2. **Dokumentasi Laravel:** Framework backend yang digunakan.
3. **Panduan React.js:** Untuk pengembangan antarmuka pengguna.
4. **OWASP Top 10:** Panduan keamanan aplikasi web.

**2. Desain Arsitektur**

**2.1 Diagram Arsitektur Sistem**

Sistem menggunakan arsitektur **MVC (Model-View-Controller)** dengan teknologi:

1. **Backend:** Laravel (PHP) untuk logika bisnis dan RESTful API.
2. **Frontend:** React.js dengan Bootstrap 5 untuk antarmuka responsif.
3. **Database:** MySQL untuk penyimpanan data akademik.
4. **Integrasi:** API dengan sistem eksternal (pembayaran, nilai).
5. **Caching:** Redis untuk optimasi performa query data yang sering diakses.

**3. Desain Modul dan Komponen**

**3.1 Modul Autentikasi dan Otorisasi**

1. **Fungsi:** Login, logout, manajemen sesi, dan kontrol akses berbasis peran.
2. **Input:** Email/NIM, password, kode OTP (opsional).
3. **Output:** Token JWT untuk autentikasi, akses sesuai peran pengguna.
4. **Teknologi:** Laravel Sanctum untuk autentikasi API, RBAC (Role-Based Access Control).

**3.2 Modul Registrasi Mata Kuliah**

1. **Fungsi:** Pemilihan mata kuliah dengan validasi prasyarat, kapasitas kelas, dan jadwal.
2. **Input:** Pilihan mata kuliah, data mahasiswa.
3. **Output:** Status registrasi (berhasil/gagal), jadwal terbaru.
4. **Teknologi:** Algoritma pengecekan prasyarat (GraphQL query), Redis untuk caching data jadwal.

**3.3 Modul Laporan dan Statistik**

1. **Fungsi:** Menghasilkan laporan jumlah peserta per kelas, kuota tersisa, dan statistik registrasi.
2. **Output:** File PDF/Excel, dashboard visualisasi data.
3. **Teknologi:** Laravel Excel untuk ekspor data, Chart.js untuk grafik interaktif.

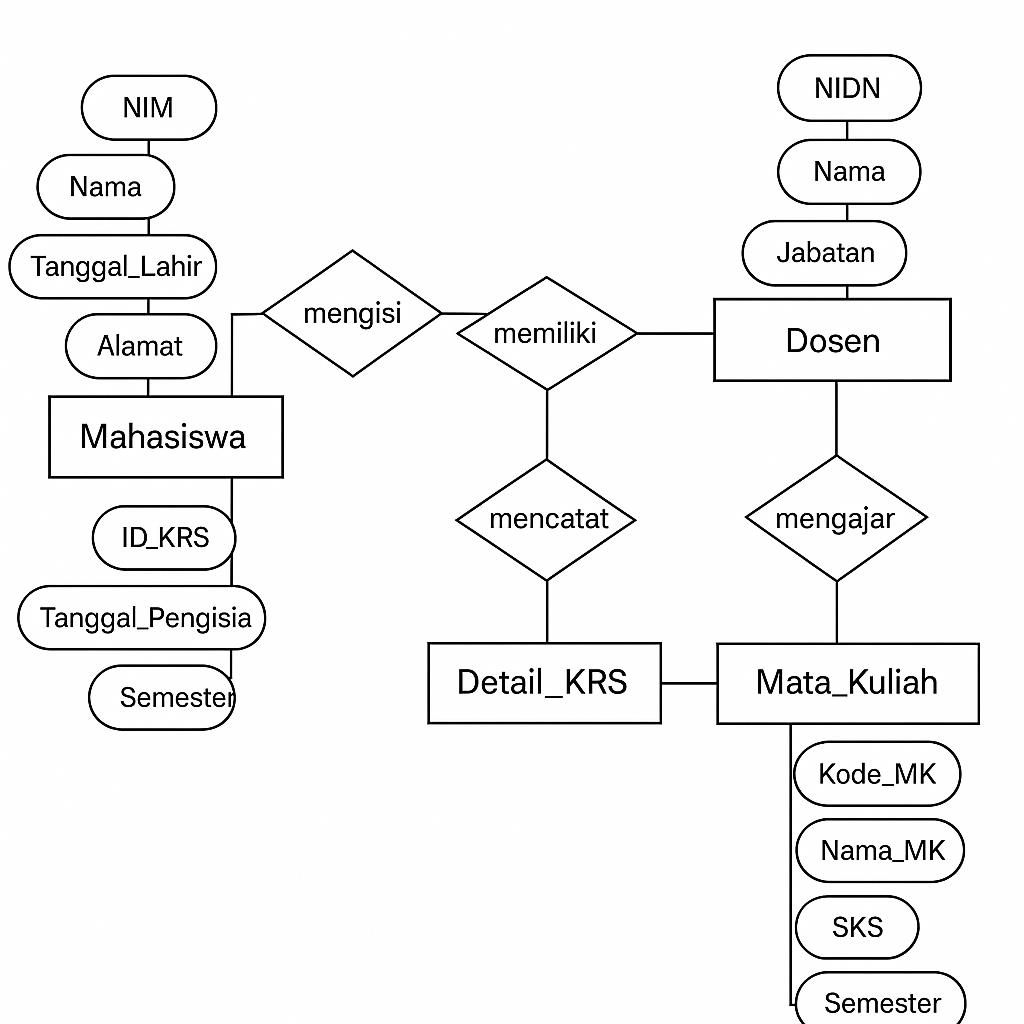
**4. Desain Basis Data**

**4.1 Model Data**

Tabel utama:

1. **mahasiswa** (nim, nama, program\_studi, angkatan, status\_pembayaran).
2. **mata\_kuliah** (kode\_mk, nama\_mk, sks, prasyarat, kapasitas, semester).
3. **registrasi** (id, nim, kode\_mk, tanggal\_registrasi, status).
4. **jadwal** (kode\_mk, hari, jam\_mulai, jam\_selesai, ruangan).
5. **prasyarat** (kode\_mk, prasyarat\_kode\_mk).

**4.2 Diagram Entity-Relationship (ERD)**



**4.3 Optimasi Database**

1. **Indexing** pada kolom nim, kode\_mk, dan tanggal\_registrasi.
2. **Partisi tabel registrasi** berdasarkan semester untuk optimasi query.
3. **Backup otomatis** setiap hari menggunakan Laravel Scheduler.

**5. Antarmuka Pengguna**

**5.1 Wireframe Desain**

1. **Halaman Login:** Form input NIM/email dan password dengan opsi "Lupa Password".
2. **Dashboard Mahasiswa:** Tampilkan daftar mata kuliah yang tersedia, jadwal, dan status registrasi.
3. **Form Registrasi:** Pilih mata kuliah dengan dropdown dan validasi real-time.
4. **Halaman Admin:** Laporan statistik registrasi dan manajemen kapasitas kelas.

**5.2 Desain Responsif**

1. **Menggunakan Bootstrap 5** untuk tampilan responsif di desktop dan mobile.
2. **AJAX** untuk memperbarui data tanpa reload halaman (e.g., cek kuota kelas).
3. **Dark/Light Mode**: Opsi tema untuk kenyamanan pengguna.

**6. Pertimbangan Keamanan**

1. **Enkripsi Data:** Data sensitif (password, token) dienkripsi dengan AES-256.
2. **Kontrol Akses:** RBAC untuk membatasi akses admin, mahasiswa, dan dosen.
3. **Proteksi API:** Rate limiting dan autentikasi JWT untuk mencegah abuse.
4. **Audit Log:** Mencatat aktivitas registrasi, perubahan data, dan login.
5. **Firewall:** Pencegahan SQL Injection dan XSS dengan middleware Laravel.

**7. Pengujian dan Validasi**

**7.1 Pengujian Unit**

1. Pengujian validasi prasyarat menggunakan PHPUnit.
2. Simulasi skenario registrasi dengan jadwal bertabrakan.

**7.2 Pengujian Keamanan**

1. Uji penetrasi dengan OWASP ZAP untuk mengidentifikasi kerentanan.
2. Pengujian autentikasi ganda (2FA) dengan OTP.

**7.3 Uji Fungsionalitas**

1. Pengujian integrasi API dengan sistem pembayaran menggunakan Postman.
2. Uji performa dengan 1000+ pengguna simultan menggunakan JMeter.

**8. Lampiran**

1. **Diagram Kelas:** Menunjukkan hubungan antar kelas dalam sistem.
2. **Dokumentasi API:** Endpoint untuk integrasi dengan sistem eksternal.
3. **Laporan Pengujian:** Hasil pengujian unit, keamanan, dan performa.
4. **Daftar Kebutuhan Tambahan:** Fitur AI untuk rekomendasi mata kuliah (rencana masa depan).