**Dokumen Desain Perangkat Lunak (Software Design Document - SDD)**

## **Pendahuluan**

## **1.1 Tujuan**

Dokumen ini mendeskripsikan desain teknis modul Kartu Rencana Studi (KRS) dalam sistem akademik berbasis web, termasuk pengisian KRS oleh mahasiswa, validasi oleh dosen wali, dan monitoring peserta mata kuliah oleh admin.

## **1.2 Ruang Lingkup**

Modul ini mencakup:

* Pengisian dan perubahan KRS oleh mahasiswa.
* Persetujuan atau penolakan KRS oleh dosen wali.
* Monitoring jumlah peserta setiap mata kuliah oleh admin.
* **Referensi**



# **Desain Arsitektur**

## **Diagram Arsitektur Sistem**

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

## **Backend**: **Yii2 Framework (PHP)**

## **Frontend**: **Bootstrap 5**

## **Database**: **MySQL**

## **API**: **RESTful API**

## **Caching**: **Redis untuk mempercepat pengambilan data KRS dan jadwal**

## **Komponen Utama Sistem**

1. **Model**: Data mahasiswa, KRS, mata kuliah
2. **View**: UI KRS (form pengisian, validasi, status)
3. **Controller**: Logika bisnis (cek SKS, IPK, validasi dosen)
4. **API**: Integrasi ke sistem nilai, keuangan, jadwal
5. **Redis**: Cache data mata kuliah & quota

# **Desain Modul dan Komponen**

## **Modul Autentikasi Pengguna**

## Fungsi**: Pengisian, perubahan, pengajuan, validasi, dan monitoring KRS oleh mahasiswa, dosen wali, dan admin.**

## Input**: NIM mahasiswa, semester aktif, daftar mata kuliah, status pengajuan, status validasi dosen wali.**

## Outpu**t: Data KRS mahasiswa per semester, status KRS (draf, diajukan, disetujui, ditolak), laporan jumlah peserta mata kuliah.**

## Teknolog**i: Yii2 Framework berbasis MVC, MySQL untuk penyimpanan data, Redis untuk cache jadwal dan kuota, RESTful API untuk integrasi dengan sistem lain.**

## Validasi**: Maksimal SKS berdasarkan IPK, pengecekan bentrok jadwal, pengecekan prasyarat mata kuliah, serta pengecekan kuota mata kuliah.**

## **Modul Manajemen Data**

* **Fungsi**: CRUD untuk entitas data KRS, mahasiswa, mata kuliah, dan validasi dosen wali.
* **Input**: Data mahasiswa (NIM, IPK, semester), data mata kuliah (kode, nama, SKS, jadwal, kuota), data KRS (daftar matkul terpilih, status pengajuan), dan data dosen wali.
* **Output**: Data KRS per semester, status validasi, data jumlah peserta per mata kuliah, riwayat perubahan KRS.
* **Teknologi**: ORM Yii2 untuk interaksi database KRS, indexing kuota dan jadwal untuk mempercepat proses validasi dan monitoring, serta penyimpanan terstruktur dalam MySQL.

## **Modul Laporan dan Analitik**

* **Fungsi:** Pembuatan laporan data KRS, rekap jumlah peserta per mata kuliah, SKS yang diambil mahasiswa, dan statistik validasi.
* **Output:** Laporan dalam bentuk file PDF atau Excel, grafik jumlah peminat per mata kuliah, grafik distribusi SKS mahasiswa.
* **Teknologi:** Library MPDF untuk ekspor PDF, PhpSpreadsheet untuk ekspor Excel, Chart.js untuk visualisasi data rekap KRS dan statistik.

## Modul Integrasi API

* **Fungsi:** Komunikasi data KRS dengan sistem akademik lainnya, seperti sistem nilai, sistem pembayaran UKT, dan sistem presensi.
* **Input:** Data KRS mahasiswa dalam format JSON (NIM, semester, daftar mata kuliah).
* **Output:** Respons dari layanan eksternal seperti validasi status keuangan, sinkronisasi data nilai, atau pembaruan status presensi.
* **Teknologi:** Implementasi RESTful API dengan autentikasi berbasis API key, dukungan JSON untuk pertukaran data lintas sistem

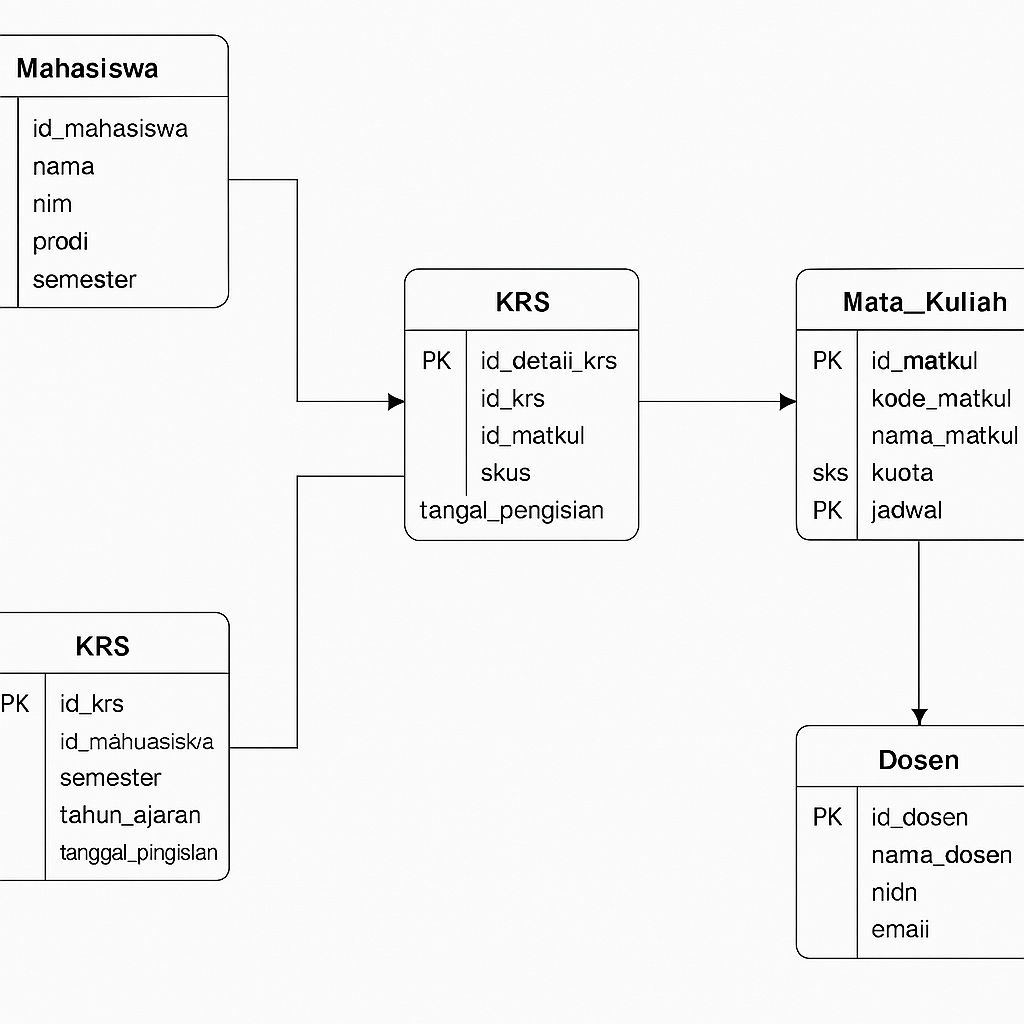
# **Desain Basis Data**

## **Model Data**

Struktur utama basis data mencakup tabel:

1. **mahasiswa** (id, nim, nama, email, prodi, semester\_aktif, ipk).
2. **mata\_kuliah** (id, kode, nama, sks, kuota, jadwal, prasyarat).
3. **krs** (id, mahasiswa\_id, semester, status\_pengajuan, status\_validasi, tanggal\_pengajuan).
4. **krs\_detail** (id, krs\_id, mata\_kuliah\_id, status\_mk).
5. **dosen\_wali** (id, nama, nip, email, fakultas).
6. **validasi\_krs** (id, krs\_id, dosen\_id, status, tanggal\_validasi).

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



## **Skema Indexing dan Optimasi**

* **Indexing** pada kolom nim mahasiswa, kode mata kuliah, dan semester untuk meningkatkan performa query KRS.
* **Partisi** pada tabel krs berdasarkan semester untuk efisiensi pengambilan data historis.
* **Backup otomatis setiap 24 jam** untuk menjamin pemulihan data jika terjadi kesalahan sistem atau kehilangan data.

# Antarmuka Pengguna

## **Mahasiswa**:

* Form pengisian KRS
* Daftar mata kuliah yang tersedia
* Status pengajuan (menunggu/pending/disetujui).

 **Dosen Wali**:

* Daftar mahasiswa
* Halaman validasi KRS
* Opsi komentar/revisi

 **Admin**:

* Monitoring kuota
* Statistik pengambilan mata kuliah
* Grafik jumlah peserta

# Pertimbangan Keamanan

* **Enkripsi Data**: Menggunakan AES-256 untuk menyimpan data sensitif.
* **Kontrol Akses**: Implementasi Role-Based Access Control (RBAC) untuk membatasi hak akses pengguna.
* **Proteksi API**: Penggunaan API key dan token autentikasi untuk mencegah akses tidak sah.
* **Audit Log**: Penyimpanan semua aktivitas pengguna untuk keamanan dan kepatuhan.
* **Firewall Aplikasi Web (WAF)**: Mencegah serangan SQL Injection dan XSS.

# Pengujian dan Validasi

**• Pengujian Unit**

* Pengujian setiap modul sistem KRS seperti pengisian, perubahan, dan validasi KRS menggunakan **PHPUnit**.
* Simulasi berbagai **skenario input**: mata kuliah tersedia, sudah penuh, prasyarat belum terpenuhi, hingga pengisian di luar jadwal.

**• Pengujian Keamanan**

* **Pengujian penetrasi** terhadap endpoint API KRS untuk mengidentifikasi potensi celah keamanan, seperti manipulasi data pengisian KRS.
* **Uji beban (stress test)** untuk memastikan sistem tetap responsif saat periode puncak pengisian KRS (awal semester).

**• Uji Fungsionalitas**

* Pengujian semua fitur utama KRS: pengisian, penyimpanan, validasi dosen wali, dan pengecekan jadwal bentrok sesuai spesifikasi kebutuhan.
* Penggunaan **Selenium** untuk mengotomatisasi pengujian alur pengisian KRS melalui antarmuka pengguna (UI).

**8. Lampiran**

* Diagram **kelas dan urutan proses** pada sistem KRS (misalnya alur pengisian hingga validasi dosen).
* Dokumentasi **API KRS** untuk integrasi dengan sistem akademik lainnya (mis. sistem pembayaran atau presensi).
* **Laporan hasil pengujian** unit dan integrasi serta rekomendasi perbaikan berdasarkan pengujian fungsional dan keamanan.