**Software Requirement Specification (SRS)**

# **1. Pendahuluan**

## **1.1 Tujuan**

## **Dokumen ini menjelaskan spesifikasi kebutuhan perangkat lunak untuk sistem Kartu Rencana Studi (KRS) yang digunakan oleh mahasiswa untuk melakukan pengisian mata kuliah per semester, dan oleh dosen serta admin untuk validasi dan pengelolaan data akademik.**

## **1.2 Ruang Lingkup**

Sistem KRS ini merupakan bagian dari sistem informasi akademik yang memungkinkan:

* Mahasiswa mengisi dan mengubah KRS.
* Dosen wali memverifikasi KRS.
* Admin mengelola data mata kuliah, jadwal, dan kuota kelas.

## **1.3 Definisi, Akronim, dan Singkatan**

* **KRS**: Kartu Rencana Studi
* **SIAKAD**: Sistem Informasi Akademik
* **SKS**: Satuan Kredit Semester
* **NIM**: Nomor Induk Mahasiswa

## **1.4 Referensi**

* IEEE Std 830-1998: Standard for Software Requirements Specification.
* Dokumentasi Yii2 Framework.
* Standar keamanan data ISO 27001.

## **1.5 Gambaran Umum Dokumen**

Dokumen ini terdiri dari pendahuluan, deskripsi umum sistem, spesifikasi kebutuhan fungsional dan non-fungsional, serta batasan sistem.

# **2. Deskripsi Umum**

**2.1 Perspektif Produk**

Sistem ini merupakan aplikasi web yang terintegrasi dengan database akademik kampus. Aplikasi ini diakses oleh mahasiswa, dosen, dan admin akademik.

## **2.2 Fungsi-Fungsi Produk**

 Mahasiswa dapat melihat jadwal dan mengisi KRS.

 Dosen dapat menyetujui atau menolak KRS mahasiswa.

 Admin dapat mengatur jadwal kuliah, dosen pengampu, dan kuota kelas.

## **2.3 Karakteristik Pengguna**

 **Mahasiswa**: Pengguna utama, mengisi KRS setiap awal semester.

 **Dosen Wali**: Menyetujui KRS mahasiswa bimbingannya.

 **Admin Akademik**: Mengelola data dan jadwal perkuliahan.

## **2.4 Batasan Sistem**

* Sistem hanya bisa diakses oleh pengguna yang sudah login.
* Jadwal KRS dibuka hanya pada waktu tertentu.
* Maksimal SKS diatur sesuai IPK mahasiswa.

# **3. Kebutuhan Spesifik**

## **3.1 Kebutuhan Fungsional**

* **3.1 Autentikasi Pengguna**
* **F01**: Pengguna dapat login menggunakan NIM/NIP dan password.
* **F02**: Sistem memberikan hak akses sesuai peran pengguna.
* **3.2 Pengisian KRS (Mahasiswa)**
* **F03**: Mahasiswa dapat memilih mata kuliah sesuai semester dan IPK.
* **F04**: Mahasiswa dapat menyimpan dan mengirimkan KRS untuk disetujui.
* **3.3 Persetujuan KRS (Dosen Wali)**
* **F05**: Dosen dapat melihat KRS mahasiswa bimbingan.
* **F06**: Dosen dapat menyetujui atau menolak pengajuan KRS.
* **3.4 Manajemen Data Akademik (Admin)**
* **F07**: Admin dapat menambah, mengubah, atau menghapus data mata kuliah.
* **F08**: Admin dapat mengatur jadwal dan kuota kelas.

## **3.2 Kebutuhan Non-Fungsional**

* **NF01**: Sistem dapat diakses 24/7 kecuali saat maintenance.
* **NF02**: Sistem mendukung browser modern (Chrome, Firefox).
* **NF03**: Waktu respon tidak lebih dari 2 detik per permintaan.
* **NF04**: Data mahasiswa dan KRS harus dienkripsi dalam database.

## **3.3 Antarmuka Pengguna**

* Halaman login
* Dashboard mahasiswa (lihat IPK, jadwal, status KRS)
* Form pengisian KRS
* Dashboard dosen wali (daftar mahasiswa bimbingan)
* Halaman admin (manajemen mata kuliah, jadwal)

**3.4 Antarmuka Perangkat Keras**

Sistem KRS membutuhkan infrastruktur server dengan spesifikasi minimal sebagai berikut:

* **Server Aplikasi**:
  + RAM: 8 GB
  + CPU: Quad-Core (minimal 2.4 GHz)
  + Penyimpanan: SSD 100 GB
  + Koneksi: Minimal 100 Mbps untuk akses internal dan eksternal

## **3.5 Antarmuka Perangkat Lunak**

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

 **Database**: MySQL versi 5.7 ke atas

 **Web Server**: Apache atau Nginx

 **API**: RESTful API digunakan untuk integrasi sistem akademik lainnya, termasuk:

* Sistem Informasi Nilai (untuk sinkronisasi prasyarat mata kuliah)
* Sistem Keuangan (untuk validasi status pembayaran)
* Sistem Monitoring Kehadiran (untuk rekap peserta aktif)

 **Autentikasi**: Setiap interaksi API diamankan dengan token JWT

 **Validasi Data**: Validasi otomatis saat mahasiswa memilih mata kuliah (IPK, SKS maksimum, konflik jadwal)

# **4. Lampiran**

**4.1 Diagram Arsitektur Sistem**

Diagram ini menggambarkan arsitektur tiga lapis (three-tier architecture):

* **Frontend Layer** (web UI mahasiswa/dosen/admin)
* **Application Layer** (Yii2 API dan business logic KRS)
* **Data Layer** (Database MySQL untuk data KRS, mahasiswa, dan jadwal kuliah)

**4.2 Contoh Wireframe Tampilan Antarmuka**

Terdiri dari:

* Halaman login pengguna
* Halaman pengisian dan pengubahan KRS oleh mahasiswa
* Tampilan status pengajuan KRS
* Halaman validasi KRS oleh dosen wali
* Dashboard admin untuk melihat kuota dan jumlah peserta per kelas

**4.3 Daftar API Endpoint untuk Integrasi Eksternal**

| **o** | **Endpoint** | **Method** | **Deskripsi** |
| --- | --- | --- | --- |
| **1** | /api/login | POST | Login mahasiswa/dosen/admin |
| **2** | /api/krs/view | GET | Melihat KRS aktif mahasiswa |
| **3** | /api/krs/edit | PUT | Mengubah KRS sebelum disetujui |
| **4** | /api/krs/submit | POST | Menyimpan dan mengirim KRS untuk persetujuan |
| **5** | /api/krs/approve | POST | Validasi dan persetujuan oleh dosen wali |
| **6** | /api/matakuliah/quota | GET | Melihat jumlah peserta dan kuota mata kuliah |