**Software Requirement Specification (SRS) Untuk Modul Registrasi Mata Kuliah (KRS)**

**1. Pendahuluan**

**1.1 Tujuan**

Dokumen ini bertujuan untuk mendefinisikan kebutuhan fungsional dan non-fungsional Modul Registrasi Mata Kuliah (KRS) berbasis web. Dokumen ini menjadi acuan bagi pengembang, tester, dan stakeholder untuk memastikan sistem memenuhi tujuan peningkatan efisiensi registrasi akademik.

**1.2 Ruang Lingkup**

Modul KRS mencakup fitur:

1. Registrasi mandiri oleh mahasiswa dengan validasi prasyarat dan jadwal.
2. Manajemen peran pengguna: admin, mahasiswa, dan dosen.
3. Integrasi dengan sistem nilai, pembayaran, dan jadwal akademik.
4. Laporan statistik registrasi dan kapasitas kelas.
5. Notifikasi real-time (email/UI) untuk konflik jadwal atau perubahan status registrasi.

**1.3 Definisi, Akronim, dan Singkatan**

1. KRS: Kartu Rencana Studi.
2. CRUD: Create, Read, Update, Delete.
3. RBAC: Role-Based Access Control.
4. API: Application Programming Interface.

**1.4 Referensi**

1. IEEE Std 830-1998: Standar Spesifikasi Kebutuhan Perangkat Lunak.
2. ISO 27001: Standar manajemen keamanan informasi.
3. Dokumentasi Laravel & React.js: Framework pengembangan sistem.

**1.5 Gambaran Umum Dokumen**

Dokumen ini terdiri dari:

1. Deskripsi umum sistem.
2. Spesifikasi kebutuhan fungsional dan non-fungsional.
3. Antarmuka pengguna dan sistem.
4. Lampiran diagram dan contoh tampilan.

**2. Deskripsi Umum**

**2.1 Perspektif Produk**

Modul KRS adalah subsistem terintegrasi dalam Sistem Informasi Akademik (SIA) perguruan tinggi. Dibangun dengan:

1. Backend: Laravel (PHP) untuk logika bisnis.
2. Frontend: React.js dengan Bootstrap 5 untuk antarmuka responsif.
3. Database: MySQL untuk penyimpanan data akademik.
4. API: RESTful untuk integrasi dengan sistem eksternal.

**2.2 Fungsi-Fungsi Produk**

1. Registrasi mata kuliah dengan validasi prasyarat dan kapasitas kelas.
2. Deteksi konflik jadwal otomatis.
3. Dashboard interaktif untuk mahasiswa dan admin.
4. Laporan registrasi dalam format PDF/Excel.
5. Notifikasi via email dan antarmuka sistem.

**2.3 Karakteristik Pengguna**

1. Admin: Mengelola data mata kuliah, jadwal, dan kapasitas kelas.
2. Mahasiswa: Melakukan registrasi, melihat jadwal, dan status KRS.
3. Dosen: Mengakses daftar peserta kelas yang diampu.

**2.4 Batasan Sistem**

1. Hanya dapat diakses melalui browser modern (Chrome, Firefox, Edge).
2. Memerlukan koneksi internet stabil untuk integrasi API.
3. Server minimal: RAM 16GB, CPU Octa-Core, Storage SSD 200GB.

**3. Kebutuhan Spesifik**

**3.1 Kebutuhan Fungsional**

|  |  |  |
| --- | --- | --- |
| **ID** | **Kebutuhan** | **Deksripsi** |
| F01 | Registrasi Pengguna | Mahasiswa/admin/login dengan NIM dan password |
| F02 | Validasi Prasyarat | Sistem memeriksa kelayakan mata kuliah berdasarkan nilai prasyarat |
| F03 | Deteksi Konflik Jadwal | Peringatan otomatis jika jadwal kuliah bertabarakan |
| F04 | Pembuatan Laporan | Admin dapat ekspor data registrasi dalam PDF/Excel |
| F05 | Notifikasi Real-Time | Mahasiswa mendapat notifikasi email saat registrasi berhasil/gagal |

**3.2 Kebutuhan Non-Fungsional**

|  |  |  |
| --- | --- | --- |
| ID | Kebutuhan | Deskripsi |
| NF01 | Perfoma | Waktu respons maksimal 2 detik untuk 1000 pengguna bersamaan |
| NF02 | Keamanan | Enkripsi AES-256 untuk data sensitif dan RBAC |
| NF03 | Ketersediaan | Uptime 99.9% dengan mekanisme failover server |
| NF04 | Kompatibilitas | Antarmuka responsif di desktop, tablet, dan mobile |

**3.3 Antarmuka Pengguna**

* **Halaman Login:** Form NIM dan password dengan opsi pemulihan akun.
* **Dashboard Mahasiswa:**

1. Daftar mata kuliah tersedia + filter semester.
2. Jadwal kuliah dan indikator konflik.
3. Tombol "Submit KRS" dengan pratinjau jadwal.

* **Halaman Admin:**

1. Tabel registrasi dengan filter program studi.
2. Grafik kapasitas kelas dan statistik registrasi.

**3.4 Antarmuka Perangkat Keras**

**Server:**

1. RAM 16GB, CPU Octa-Core, SSD 200GB.
2. Jaringan: 1 Gbps untuk menangani traffic tinggi.

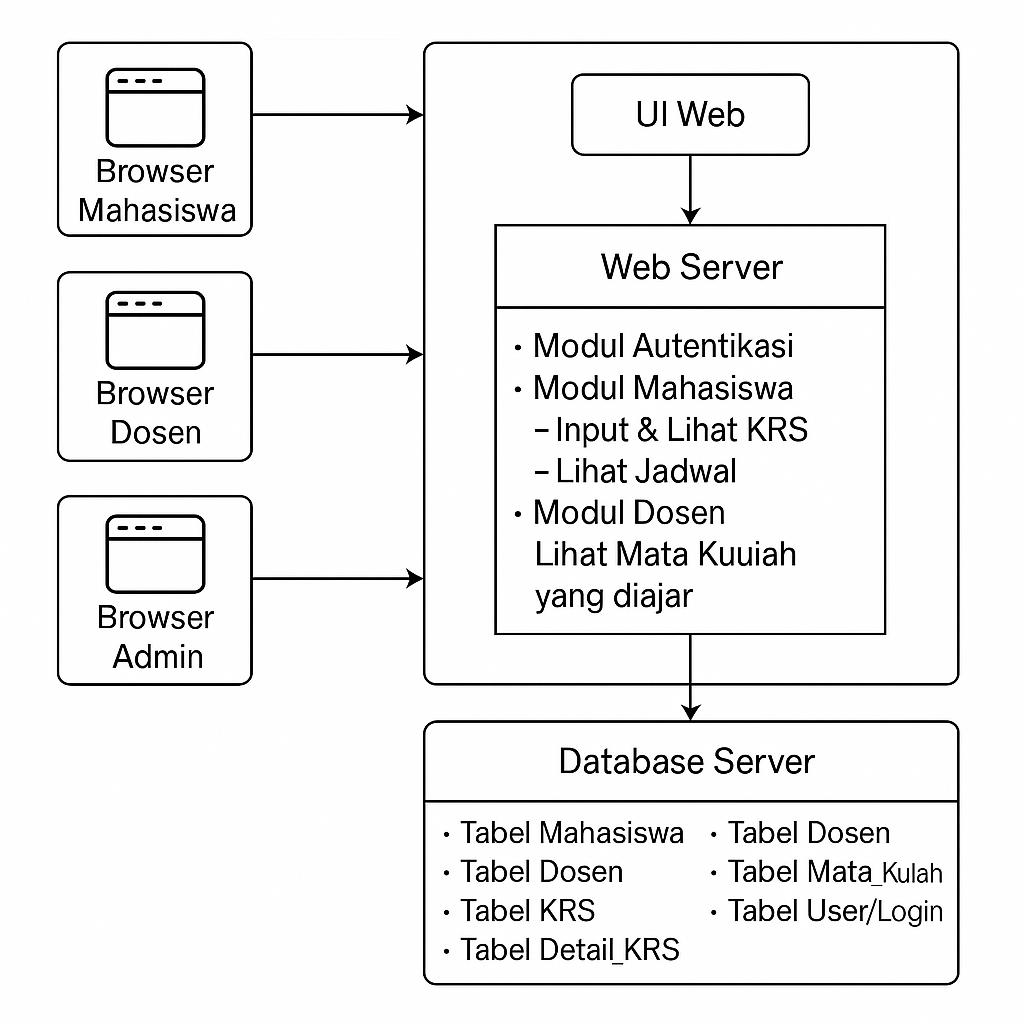
**Client:** Browser modern dengan resolusi minimal 1280x720.

**3.5 Antarmuka Perangkat Lunak**

1. Backend: Laravel 10, PHP 8.2.
2. Frontend: React.js 18, Bootstrap 5.
3. Database: MySQL 8.0 dengan partisi tabel.
4. API: RESTful untuk integrasi sistem nilai dan pembayaran.

**4. Lampiran**

**Diagram Arsitektur Sistem:**



**Wireframe Antarmuka:**

* Halaman Registrasi:

Wireframe Registrasi



**Daftar API Endpoint:**

1. POST /api/registrasi: Submit data registrasi.
2. GET /api/jadwal/{nim}: Ambil jadwal mahasiswa.
3. GET /api/laporan: Generate laporan registrasi (admin).