**SOFTWARE REQUIREMENTS SPECIFICATION**

**Di susun 12 April 2025**

**Kelompok 1**

**Sukran Golit ( B02220100 )**

**Maimanatul Karimah ( B02220122 )**

**Jurusan Ilmu Komputer  
Universitas Muhammadiyah BIMA  
2025/2026**

**1. PENDAHULUAN**

**1.1 Tujuan**

Tujuan dibuatnya dokumen SRS ini adalah untuk menjelaskan tentang spesifikasi kebutuhan apa saja yang digunakan pada Manajemen Data Mahasiswa seperti Pendaftaran mahasiswa baru, Pengelolaan biodata mahasiswa, Status mahasiswa (aktif, cuti, DO, alumni).

**1.2 Lingkup Masalah**

Lingkup masalah manajemen data mahasiswa untuk website meliputi Manajemen Pengguna dengan Autentikasi dan Otorisasi, Pengolahan dan Penyimpanan Data yang Aman, Laporan dan Analitik dan Integrasi dengan Sistem Lain melalui API.

1. Manajemen Pengguna dengan Autentikasi dan Otorisasi

Fitur ini memastikan bahwa hanya pengguna yang memiliki hak akses yang sesuai yang dapat masuk dan menggunakan sistem.

Autentikasi: Proses verifikasi identitas pengguna, biasanya melalui username dan password, atau metode lain seperti OTP atau autentikasi dua faktor (2FA).

Otorisasi: Setelah pengguna terverifikasi, sistem menentukan apa saja yang boleh dan tidak boleh dilakukan berdasarkan peran atau level akses (misalnya admin, pengguna biasa, atau supervisor).

2. Pengolahan dan Penyimpanan Data yang Aman dan Terstruktur

Data yang masuk ke sistem akan diolah, divalidasi, dan disimpan dalam basis data dengan struktur yang rapi.

Keamanan: Meliputi enkripsi data, proteksi terhadap akses ilegal, serta backup data secara rutin.

Terstruktur: Data disimpan dalam format dan skema tertentu agar mudah dicari, digunakan, dan dianalisis.

3. Laporan dan Analitik Berbasis Data

Sistem dapat menghasilkan laporan dari data yang dikumpulkan untuk membantu pemantauan, evaluasi, dan pengambilan keputusan.

Bisa berupa laporan harian, mingguan, atau bulanan.

Analitik bisa mencakup visualisasi data seperti grafik, diagram, atau statistik penting.

4. Integrasi dengan Sistem Lain melalui API

Fitur ini memungkinkan sistem untuk berkomunikasi atau bertukar data dengan aplikasi atau platform lain.

API (Application Programming Interface) adalah antarmuka yang memungkinkan sistem-sistem berbeda saling terhubung.

**1.3 Definisi, Akronim, Singkatan**

CRUD: Create, Read, Update, Delete

CRUD adalah empat fungsi dasar yang digunakan dalam pengelolaan data di sistem basis data maupun aplikasi.

**Create** : Menambahkan data baru.

**Read** : Membaca atau menampilkan data yang sudah ada.

**Update** : Memperbarui atau mengedit data yang sudah ada.

**Delete** : Menghapus data yang tidak diperlukan.

2. API: Application Programming Interface

API menyediakan cara standar bagi sistem lain untuk mengakses fitur atau data tanpa harus tahu bagaimana sistem itu dibangun.

3. RBAC: Role-Based Access Control

RBAC adalah metode pengelolaan hak akses pengguna berdasarkan peran (role) mereka dalam sistem.

Setiap role punya hak akses tertentu.

Contohnya:

* Admin bisa akses dan ubah semua data.
* Manager bisa melihat semua laporan tapi tidak bisa menghapus data.
* User hanya bisa lihat dan ubah data milik sendiri.

RBAC membantu menjaga keamanan sistem dengan memastikan bahwa hanya pengguna yang sesuai yang bisa mengakses fitur tertentu.

**1.4 Referensi**

* IEEE. IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications. IEEE Computer Society, 1998.
* Standar Keamanan Data: ISO/IEC 27001

Dalam pengelolaan data mahasiswa, keamanan informasi menjadi aspek yang sangat penting. Sistem ini dirancang dengan mengacu pada standar internasional ISO/IEC 27001, yang merupakan standar untuk Sistem Manajemen Keamanan Informasi (ISMS). Standar ini menetapkan kerangka kerja yang sistematis dalam mengelola informasi sensitif, agar tetap aman melalui penerapan kebijakan, prosedur, kontrol teknis, dan organisasi.

Beberapa prinsip utama yang diadopsi dari ISO/IEC 27001 meliputi:

1. Kerahasiaan: Data mahasiswa hanya dapat diakses oleh pihak yang berwenang.
2. Integritas: Data dijaga agar tetap akurat dan tidak mengalami perubahan tanpa izin.
3. Ketersediaan: Data tersedia saat dibutuhkan oleh pengguna yang berwenang.

Penerapan standar ini bertujuan untuk melindungi data pribadi mahasiswa dari ancaman seperti kebocoran data, akses tidak sah, dan kerusakan sistem.

**1.5 Gambaran Umum**

Dokumen ini mencakup beberapa bagian utama yang menjelaskan sistem manajemen data mahasiswa. Bagian-bagian tersebut meliputi:

Pendahuluan : yang memberikan latar belakang dan tujuan dari pengembangan sistem.

Deskripsi Umum Sistem : yang menjelaskan konteks sistem, fungsi utama, serta pemangku kepentingan terkait.

Spesifikasi Kebutuhan Fungsional dan Non-Fungsional : yang merinci fitur yang harus dimiliki sistem serta kriteria teknis yang harus dipenuhi.

Batasan Sistem : yang menjelaskan ruang lingkup dan keterbatasan sistem yang akan dikembangkan.

**2. Deskripsi Umum**

**2.1 Perspektif Produk**

Sistem pengelolaan sampah berbasis teknologi ini akan dikembangkan sebagai aplikasi berbasis web yang dapat diakses melalui browser modern. Aplikasi ini dirancang untuk memberikan solusi efisien dalam pengelolaan sampah, dengan fokus pada pemisahan, pengumpulan, dan daur ulang sampah. Berikut adalah rincian lebih lanjut mengenai perspektif produk:

**Arsitektur Sistem:**

* + Client-Server: Sistem ini akan menggunakan arsitektur client-server, di mana aplikasi frontend (client) berinteraksi dengan backend (server) untuk mengelola data dan logika bisnis.
  + Backend: Backend akan dibangun menggunakan Yii2 Framework, yang merupakan framework PHP yang kuat dan efisien untuk pengembangan aplikasi web. Yii2 menyediakan berbagai fitur seperti manajemen database, keamanan, dan pengelolaan sesi yang akan mendukung pengembangan sistem.
  + Database: MySQL akan digunakan sebagai sistem manajemen basis data (DBMS) untuk menyimpan dan mengelola data. Database ini akan menyimpan informasi terkait pengguna, data sampah, jadwal pengumpulan, dan statistik daur ulang.

**2.2 Fungsi-Fungsi Produk**

Produk ini memiliki beberapa fungsi utama yang dirancang untuk menunjang pengelolaan data mahasiswa secara efektif dan efisien, yaitu:

1. **Manajemen Akun Pengguna**  
   Sistem menyediakan fitur untuk mengelola akun pengguna, termasuk administrator, dosen, dan mahasiswa. Pengguna dapat melakukan registrasi, login, pengaturan profil, serta pengelolaan hak akses sesuai peran masing-masing.
2. **Pengolahan Data Melalui Fitur CRUD**  
   Sistem mendukung fungsi **Create, Read, Update, dan Delete (CRUD)** untuk berbagai data mahasiswa seperti identitas pribadi, data akademik, riwayat studi, dan dokumen pendukung lainnya.
3. **Penyajian Laporan dan Analitik**  
   Sistem menyediakan fitur pembuatan laporan yang komprehensif, seperti rekap data mahasiswa, laporan akademik, grafik statistik, dan analitik berbasis data untuk mendukung pengambilan keputusan.
4. **Sistem Notifikasi untuk Pemberitahuan Penting**  
   Sistem dilengkapi dengan fitur notifikasi yang menginformasikan pengguna mengenai hal-hal penting, seperti pengumuman akademik, batas waktu pengisian KRS, atau perubahan jadwal perkuliahan.

**2.3 Karakteristik Pengguna**

Sistem manajemen data mahasiswa dirancang untuk digunakan oleh berbagai jenis pengguna dengan peran dan hak akses yang berbeda. Karakteristik pengguna sistem ini meliputi:

1. **Administrator**  
   Pengguna dengan akses penuh terhadap sistem. Tugas utama administrator mencakup:
   * Mengelola akun pengguna (menambahkan, mengubah, atau menghapus akun).
   * Menentukan hak akses sesuai peran.
   * Melakukan pengawasan terhadap data dan aktivitas pengguna dalam sistem.
   * Menjamin keamanan dan integritas data mahasiswa.
2. **Pengguna Umum**  
   Pengguna yang mengakses sistem berdasarkan peran yang diberikan, yang dapat mencakup:
   * **Dosen**: Mengakses data mahasiswa, menginput nilai, memberikan catatan akademik, serta melihat laporan kelas.
   * **Mahasiswa**: Mengakses dan memperbarui data pribadi, melihat hasil studi, mengunduh dokumen, dan menerima notifikasi dari sistem.
   * **Staf Akademik**: Mengelola data administrasi akademik mahasiswa seperti KRS, KHS, dan surat-surat keterangan.

**2.4 Batasan Sistem**

Dalam pengembangan sistem manajemen data mahasiswa, terdapat beberapa batasan yang perlu diperhatikan agar sistem dapat berfungsi dengan optimal dan aman, yaitu:

1. **Kompatibilitas Browser**  
   Sistem hanya dapat berjalan secara optimal pada browser modern seperti **Google Chrome**, **Mozilla Firefox**, dan **Microsoft Edge**. Penggunaan browser selain yang disebutkan dapat menyebabkan gangguan tampilan antarmuka atau fungsi-fungsi tertentu tidak berjalan sebagaimana mestinya.
2. **Keamanan Data Mahasiswa**  
   Untuk menjaga kerahasiaan dan integritas data mahasiswa, sistem harus menerapkan standar keamanan yang mencakup:

* **Enkripsi data**, baik saat data disimpan di server maupun saat ditransmisikan melalui jaringan, untuk mencegah kebocoran informasi.
* **Otorisasi berbasis peran (Role-Based Access Control)**, di mana setiap pengguna (seperti admin, dosen, mahasiswa, dan staf akademik) hanya dapat mengakses data dan fitur yang sesuai dengan hak aksesnya.

**3. Kebutuhan Spesifik**

**3.1 Kebutuhan Fungsional**

Sistem manajemen data mahasiswa harus memenuhi beberapa kebutuhan fungsional utama agar dapat digunakan secara efektif oleh seluruh pengguna. Kebutuhan-kebutuhan tersebut antara lain:

* **F01: Registrasi dan Login Pengguna**  
  Pengguna (admin, dosen, mahasiswa, dan staf akademik) dapat melakukan **registrasi akun** serta **login** ke dalam sistem menggunakan kredensial yang valid.
* **F02: Pengelolaan Data dengan Fitur CRUD**  
  Sistem harus mendukung operasi **Create, Read, Update, dan Delete (CRUD)** terhadap data pengguna dan data mahasiswa. Data yang dimaksud mencakup informasi pribadi, riwayat akademik, dan dokumen pendukung.
* **F03: Penyediaan Laporan**  
  Sistem harus mampu menghasilkan laporan dalam format **PDF** dan **Excel**, seperti:
  + Rekap data mahasiswa
  + Laporan akademik per semester
  + Statistik jumlah mahasiswa berdasarkan program studi, angkatan, atau status akademik
* **F04: Pengiriman Notifikasi**  
  Sistem harus menyediakan fitur **notifikasi** yang dapat dikirim melalui:
  + **Email** kepada pengguna untuk informasi penting (misalnya pengumuman akademik, batas waktu pengisian KRS)
  + **Notifikasi dalam sistem** (in-app notification) yang muncul saat pengguna login

**3.2 Kebutuhan Non-Fungsional**

Untuk memastikan sistem manajemen data mahasiswa berjalan dengan baik, tidak hanya dibutuhkan fungsionalitas yang lengkap, tetapi juga performa, keamanan, dan kenyamanan pengguna yang memadai. Berikut ini adalah kebutuhan non-fungsional yang harus dipenuhi oleh sistem:

* **NF01: Waktu Respons Cepat**  
  Sistem harus memiliki **waktu respons kurang dari 2 detik** untuk setiap permintaan pengguna, seperti saat menampilkan data mahasiswa atau mengakses laporan.
* **NF02: Dukungan Multi-Pengguna**  
  Sistem harus mampu **mendukung minimal 100 pengguna** yang mengakses secara bersamaan tanpa mengalami penurunan performa yang signifikan.
* **NF03: Keamanan Data dengan Enkripsi Tingkat Tinggi**  
  Untuk menjaga kerahasiaan dan integritas data mahasiswa, sistem harus menerapkan **enkripsi AES-256** pada data sensitif, baik saat disimpan maupun ditransmisikan melalui jaringan.
* **NF04: Antarmuka Responsif dan Akses Mobile**  
  Antarmuka pengguna (UI) harus **responsif** dan **dapat diakses dengan baik melalui perangkat mobile**, seperti smartphone dan tablet, agar pengguna dapat mengakses sistem kapan pun dan di mana pun.

**3.3 Antarmuka Pengguna**

Antarmuka pengguna (user interface) dirancang agar intuitif, mudah digunakan, dan dapat diakses oleh berbagai jenis pengguna sesuai perannya. Beberapa komponen utama antarmuka pengguna dalam sistem manajemen data mahasiswa antara lain:

* **Halaman Login**  
  Menyediakan **form login** yang terdiri dari kolom **username** dan **password**. Halaman ini juga dapat dilengkapi dengan opsi "Lupa Password" dan tautan ke halaman registrasi (jika diperlukan).
* **Dashboard Utama**  
  Setelah login, pengguna akan diarahkan ke dashboard yang menampilkan **informasi penting secara ringkas**, seperti:
  + Jumlah total mahasiswa
  + Statistik berdasarkan program studi atau angkatan
  + Notifikasi terbaru
  + Akses cepat ke fitur-fitur utama seperti input data, laporan, atau pengaturan akun
* **Halaman Laporan**  
  Menyediakan tampilan data dalam bentuk tabel yang dapat **difilter berdasarkan kriteria tertentu**, seperti:
  + Nama mahasiswa
  + NIM
  + Program studi
  + Tahun masuk atau status akademik  
    Selain itu, pengguna juga dapat mengunduh laporan dalam format **PDF atau Excel**.

**3.4 Antarmuka Perangkat Keras**

Untuk menjamin kelancaran operasional dan kinerja sistem manajemen data mahasiswa, diperlukan perangkat keras dengan spesifikasi minimum sebagai berikut:

* **Server**
  + **RAM**: Minimal 8 GB
  + **CPU**: Minimal Quad-Core Processor
  + **Storage**: SSD minimal 100 GB untuk mendukung kecepatan akses data
  + **Koneksi Jaringan**: Stabil dan cepat, minimal 100 Mbps, terutama untuk mendukung akses multi-pengguna

Spesifikasi ini dapat disesuaikan dengan jumlah pengguna aktif dan volume data yang dikelola dalam sistem.

**3.5 Antarmuka Perangkat Lunak**

Sistem manajemen data mahasiswa dibangun dengan memanfaatkan perangkat lunak dan teknologi pendukung tertentu guna menjamin kestabilan, kemudahan pengembangan, serta kemudahan integrasi dengan sistem lainnya. Adapun antarmuka perangkat lunak yang digunakan meliputi:

* **Framework Aplikasi**  
  Sistem dikembangkan menggunakan **Yii2 Framework**, yang merupakan framework PHP berbasis Model-View-Controller (MVC) dan mendukung pengembangan aplikasi web yang cepat, aman, dan terstruktur.
* **Database Management System (DBMS)**  
  Sistem menggunakan **MySQL** sebagai basis data untuk menyimpan informasi penting seperti data mahasiswa, pengguna, riwayat akademik, dan log aktivitas.
* **Antarmuka API (Application Programming Interface)**  
  Sistem menyediakan **API RESTful** untuk memungkinkan integrasi dengan sistem lain di lingkungan kampus, seperti:
  + Sistem informasi akademik
  + Sistem keuangan mahasiswa
  + Aplikasi mobile kampus

RESTful API juga memudahkan pengembangan aplikasi pihak ketiga yang memerlukan akses terbatas ke data mahasiswa.

**4. Lampiran**

Lampiran ini berisi dokumen pendukung yang membantu dalam memahami struktur, desain, dan mekanisme kerja sistem manajemen data mahasiswa secara lebih teknis dan visual. Beberapa komponen lampiran yang disertakan antara lain:

1. **Diagram Arsitektur Sistem**  
   Diagram ini menjelaskan struktur umum sistem, termasuk hubungan antara komponen front-end, back-end, database, serta integrasi dengan sistem eksternal melalui API. Arsitektur umumnya mencakup:
   * Pengguna (Admin, Mahasiswa, Dosen)
   * Aplikasi Web (berbasis Yii2)
   * Database Server (MySQL)
   * API RESTful Layer
   * Layanan pendukung (SMTP, sistem eksternal kampus)
2. **Contoh Wireframe Tampilan Antarmuka**  
   Wireframe disertakan untuk memberikan gambaran awal tentang desain antarmuka pengguna (UI). Contoh tampilan yang disertakan antara lain:
   * Halaman Login
   * Dashboard Utama
   * Form Input Data Mahasiswa
   * Halaman Laporan dengan Filter Pencarian
3. **Daftar API Endpoint untuk Integrasi Eksternal**  
   Daftar ini mencakup endpoint RESTful API yang tersedia untuk sistem lain yang membutuhkan integrasi, lengkap dengan:
   * URL endpoint
   * Metode (GET, POST, PUT, DELETE)
   * Deskripsi fungsi masing-masing endpoint
   * Format request dan response (JSON)
   * Otorisasi (misal: token JWT)

**Contoh:**

GET /api/mahasiswa/{id}

Deskripsi: Mengambil data lengkap mahasiswa berdasarkan NIM

Respon: {

"NIM": B02220100,

"Nama": "Sukran Golit",

"Prodi": "Ilmu Komputer",

"Angkatan": 2022

}