

Nama : Rizal Wahyudin

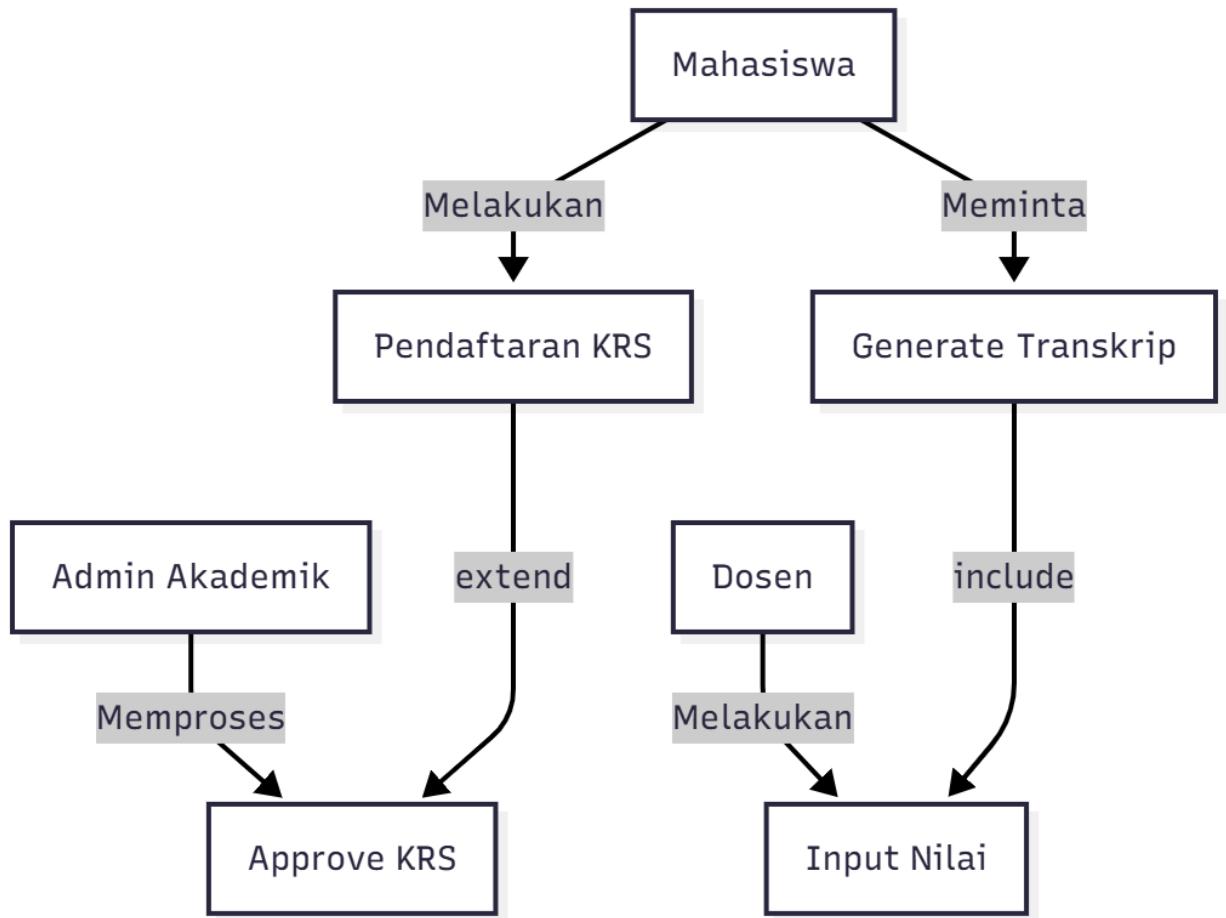
Nim : 253636004

Laporan ini menguraikan desain sistem akademik, khususnya untuk proses **Kartu Rencana Studi (KRS)** dan **Presensi**, berdasarkan diagram UML yang disediakan.

### 1. Diagram Use Case: Fungsi dan Aktor Sistem

Diagram Use Case menunjukkan fungsionalitas utama sistem dan bagaimana aktor (pengguna) berinteraksi dengannya.

- **Gambar:**



- **Penjelasan:**

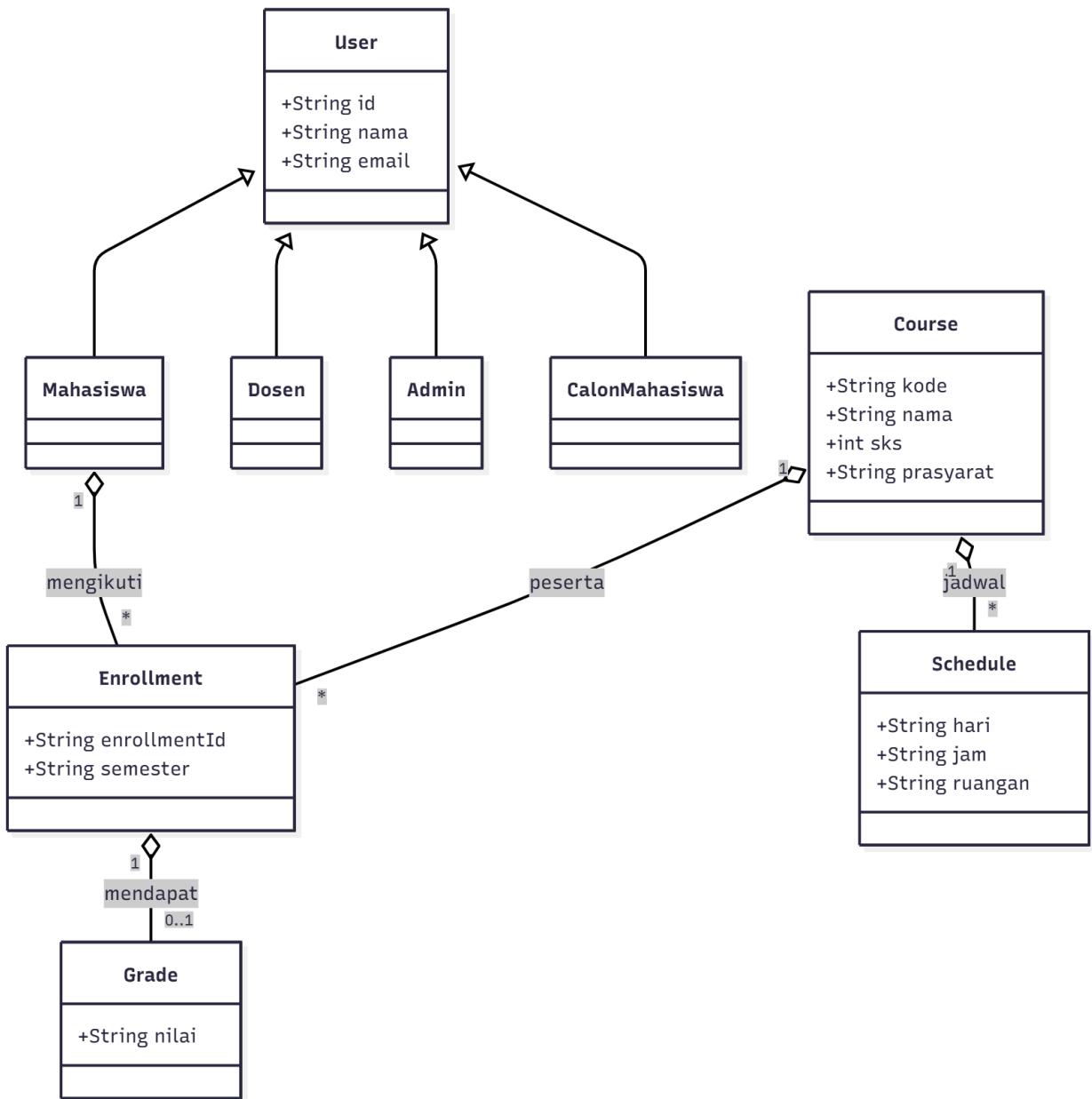
- **Aktor Utama: Mahasiswa, Dosen, Admin Akademik.**
- **Fungsi Utama:**
  - **Pendaftaran KRS:** Dilakukan oleh **Mahasiswa**.
  - **Generate Transkrip:** Diminta oleh **Mahasiswa**.
  - **Approve KRS:** Diproses oleh **Admin Akademik** (meng-*extend* Pendaftaran KRS).
  - **Input Nilai:** Dilakukan oleh **Dosen** (di-*include* dalam Generate Transkrip).
- Diagram ini memetakan **kebutuhan fungsional** sistem dari perspektif pengguna.

## 2. Diagram Kelas (Class Diagram): Struktur Data dan Hubungan

Diagram Kelas memvisualisasikan struktur statis sistem, menunjukkan kelas-kelas, atribut, operasi, dan hubungan antar kelas. Anda memiliki dua varian diagram kelas yang berbeda fokusnya.

### 2.1. Diagram Kelas Umum Sistem Akademik

- Gambar:



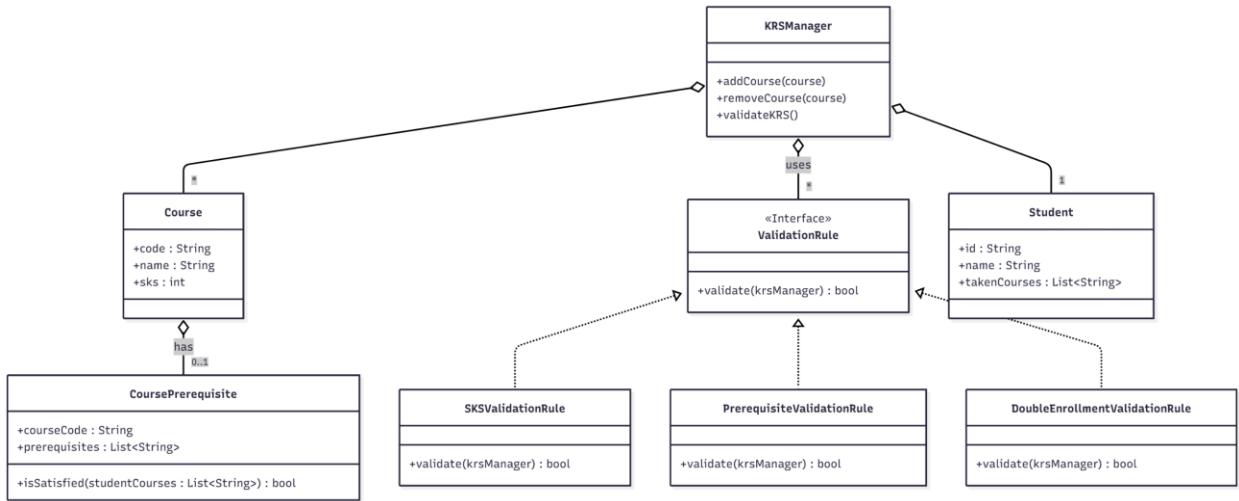
- Penjelasan:

- Menunjukkan hierarki pengguna dengan kelas **User** sebagai superclass yang diwarisi oleh **Mahasiswa**, **Dosen**, **Admin**, dan **CalonMahasiswa** (generalization/pewarisan).
- Kelas **Mahasiswa** memiliki hubungan **komposisi** (intan berlian padat) dengan **Enrollment** (1 Mahasiswa memiliki 1 Enrollment).
- Kelas **Enrollment** berasosiasi dengan **Course** dan memiliki hubungan **komposisi** dengan **Grade** (1 Enrollment mendapat 0..1 Grade).
- Kelas **Course** memiliki hubungan **agregasi** (intan berlian kosong) dengan **Schedule** (1 Course memiliki banyak Schedule).

- Diagram ini berfokus pada **struktur data dasar** pengguna, pendaftaran, mata kuliah, jadwal, dan nilai.

## 2.2. Diagram Kelas Spesifik Proses KRS (Varian A: DESAIN\_CLASS\_DIAGRAM.jpg)

- Gambar:**

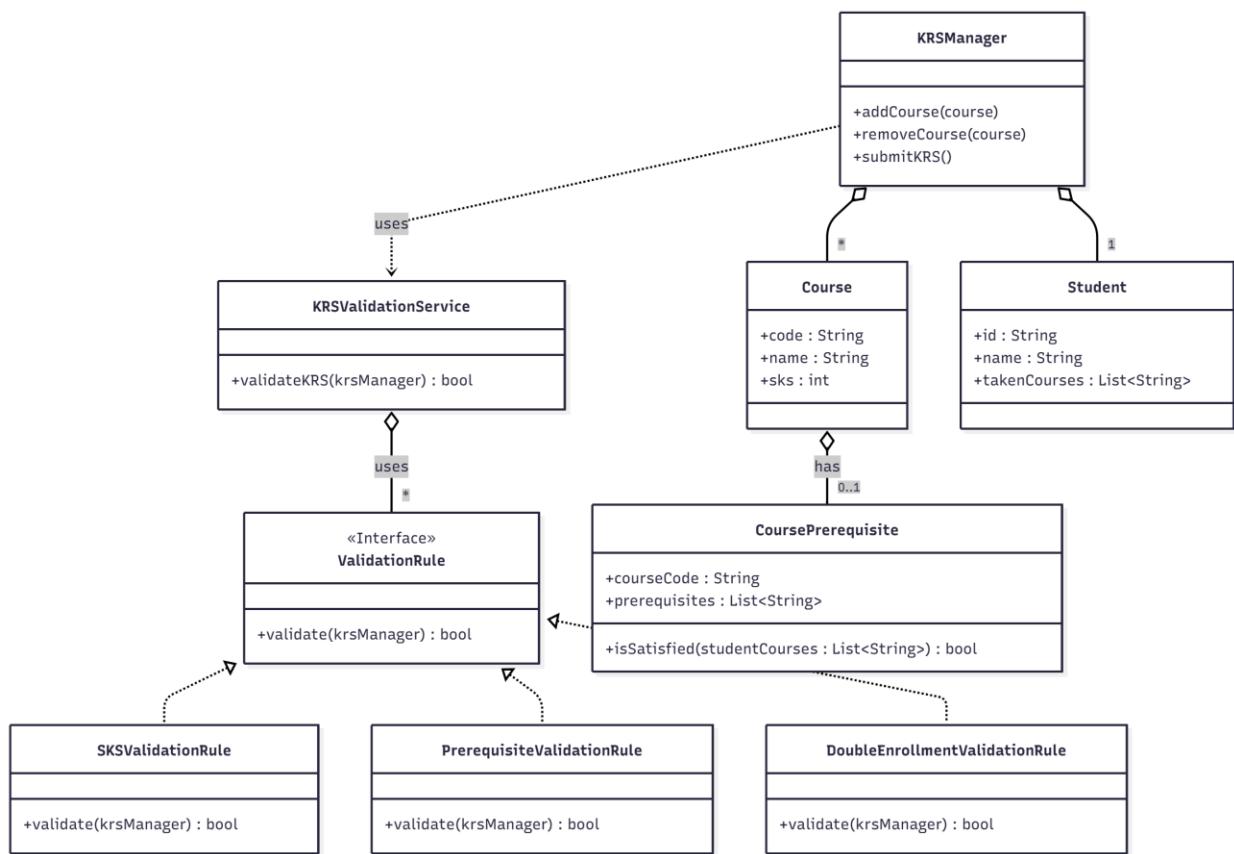


- Penjelasan:**

- Diagram ini berfokus pada **Mekanisme Validasi KRS** menggunakan pola **Strategy Pattern**.
- Kelas **KRSManager** adalah pusat pengelolaan dan validasi KRS.
- Terdapat hubungan **agregasi** (intan berlian kosong) dari **KRSManager** ke **Course** dan **Student**.
- KRSManager** memiliki hubungan ketergantungan (uses) dengan **ValidationRule** (Interface).
- Kelas-kelas seperti **SKSValidationRule**, **PrerequisiteValidationRule**, dan **DoubleEnrollmentValidationRule** mengimplementasikan (panah putus-putus ke atas) **ValidationRule** untuk menjalankan berbagai jenis validasi.
- Kelas **CoursePrerequisite** memiliki hubungan **komposisi** dengan **Course** (1 Course has 0..1 CoursePrerequisite).

### 2.3. Diagram Kelas Spesifik Proses KRS (Varian B: AUDIT\_CLASS\_DIAGRAM.jpg)

- Gambar:



- Penjelasan:

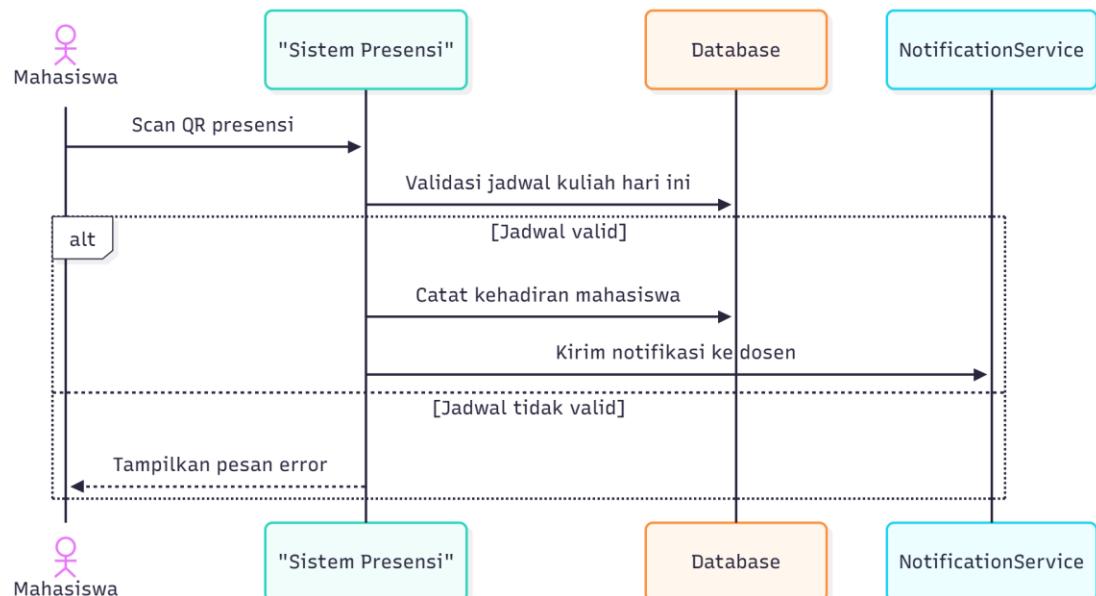
- Ini adalah pengembangan atau varian dari Diagram 2.2, menambahkan kelas **KRSValidationService**.
- **KRSManager** sekarang berinteraksi dengan **KRSValidationService** (melalui hubungan *uses*) untuk menjalankan validasi.
- **KRSValidationService** kemudian menggunakan (melalui hubungan *uses*) satu atau lebih implementasi dari **ValidationRule** (Strategy Pattern), seperti **SKSValidationRule**, **PrerequisiteValidationRule**, dan **DoubleEnrollmentValidationRule**.
- Perubahan ini menunjukkan penerapan prinsip **Single Responsibility** yang lebih kuat, di mana tugas validasi dikapsulasi dalam **KRSValidationService**.

---

### 3. Diagram Sekuens (Sequence Diagram): Interaksi Waktu Nyata

Diagram Sekuens menunjukkan interaksi objek dalam sistem berdasarkan urutan waktu untuk mencapai hasil tertentu.

- **Gambar:**



- **Penjelasan:**

- Diagram ini menggambarkan alur proses **Presensi (Absensi)**.
- **Aktor: Mahasiswa.**
- **Objek/Lifelines: "Sistem Presensi", Database, NotificationService.**
- **Alur:**
  1. **Mahasiswa** memulai dengan **Scan QR presensi**.
  2. "**Sistem Presensi**" meminta **Validasi jadwal kuliah hari ini** dari **Database**.
  3. Terdapat *fragment alt* (alternatif) untuk dua skenario:
    - **[Jadwal valid]**: Sistem mencatat kehadiran, lalu mengirim notifikasi ke dosen melalui **NotificationService**.
    - **[Jadwal tidak valid]**: Sistem menampilkan pesan *error* kepada **Mahasiswa**.
- Diagram ini merinci **perilaku dinamis** sistem selama satu kasus penggunaan spesifik.

---

Apakah Anda ingin saya memberikan detail lebih lanjut mengenai salah satu diagram atau hubungan antar kelasnya?