



## MINGGU 7-8: Modul Jadwal & Ruangan



### Deskripsi Tugas

Sistem penjadwalan kuliah dengan deteksi otomatis bentrok jadwal (hari, jam, ruangan, dosen). Implementasi Observer pattern untuk notifikasi perubahan jadwal ke stakeholder.



### Tujuan Pembelajaran

- Constraint satisfaction problem
- Observer/Pub-Sub pattern
- Conflict detection algorithm
- Calendar integration basics



### Deliverables

#### 1. Scheduling Service (35%)

- Create/Update/Delete jadwal
- Algoritma deteksi bentrok 3 dimensi (waktu-ruangan-dosen)
- Capacity check (mahasiswa enrolled  $\leq$  kapasitas ruangan)

#### 2. Observer Implementation (25%)

- NotificationManager yang notify saat jadwal berubah
- Subscriber: Mahasiswa, Dosen, Admin Akademik
- Medium: Email/SMS/Push (simulasi via log)

#### 3. Conflict Resolver UI (20%)

- Dashboard untuk admin lihat bentrok
- Suggestion alternative jadwal (AI-powered)

#### 4. Integration dengan KRS (15%)

- Saat jadwal berubah, invalidate KRS yang terpengaruh

#### 5. Testing + Documentation (5%)



### Langkah Pengerjaan

#### Step 1: Conflict Detection Algorithm

"Tulis function detect\_schedule\_conflict yang:

Input: list jadwal (hari, jam\_mulai, jam\_selesai, ruangan, dosen)

Output: list of conflict dengan detail:

- Type: 'room\_conflict', 'lecturer\_conflict', 'time\_overlap'
- Affected schedules: [schedule\_1, schedule\_2]

Algorithm: untuk setiap pair jadwal, cek:

1. Hari sama AND jam overlap AND ruangan sama → room conflict
2. Hari sama AND jam overlap AND dosen sama → lecturer conflict

Optimize dengan interval tree untuk jam overlap check."

## Step 2: Observer Pattern Implementation

# Prompt untuk AI

"""

Implementasikan Observer pattern untuk notifikasi jadwal:

# Publisher

class ScheduleSubject:

```
    def attach(observer)
    def detach(observer)
    def notify(event_type, schedule_data)
```

# Observers

class StudentObserver:

```
    def update(event) → send email ke mahasiswa terdampak
```

class LecturerObserver:

```
    def update(event) → notify dosen
```

Saat jadwal berubah, panggil subject.notify('SCHEDULE\_CHANGED', data)

"""

## Step 3: AI-Powered Rescheduling

"Anda adalah scheduling assistant. Diberikan:

- Jadwal bentrok: Kalkulus I (Senin 08:00, Ruang A101, 40 mhs)
- Available slots: (hari, jam, ruangan, kapasitas)

Rekomendasikan 3 alternatif jadwal yang:

1. Ruangan kapasitasnya cukup
2. Tidak bentrok dengan jadwal dosen
3. Minimal disruption untuk mahasiswa (prioritas pagi/siang)

Output format JSON: "[{hari, jam, ruangan, reason}, ...]"

### Kriteria Sukses

- [ ] Sistem deteksi semua jenis bentrok
  - [ ] Notifikasi terkirim saat jadwal berubah (cek log)
  - [ ] Sequence diagram Observer pattern tersedia
  - [ ] Admin mendapat saran jadwal alternatif dari AI
-