

TUGAS 1

NEGOSIASI & PEMESANAN LAYANAN DENGAN DUA AGEN

Tujuan Pembelajaran:

1. Menerapkan protokol komunikasi agen (FIPA-ACL atau KQML) secara benar.
2. Mendesain ontologi pesan (atribut, tipe data, dan semantik) untuk domain terpilih.
3. Mengimplementasikan alur negosiasi sederhana (tanya → tawar/opsi → pilih → pesan → konfirmasi).
4. Mengelola keadaan (*state*) agen dan log percakapan untuk evaluasi.

Spesifikasi Teknis:

1) Alur Minimum (*Happy Path* + *Error Path*)

Happy Path (wajib):

1. Customer → Provider: permintaan informasi (ketersediaan/opsi/slot/harga).
2. Provider → Customer: detail opsi (terstruktur).
3. Customer → Provider: permintaan pemesanan (spesifik).
4. Provider → Customer: konfirmasi & ringkasan transaksi.

Error Path (wajib, pilih minimal 1 kondisi):

- Opsi tidak tersedia / stok habis.
- Kursi sudah terpesan

2) Protokol Komunikasi (pilih salah satu: FIPA-ACL atau KQML)

a. FIPA-ACL (*contoh skema*)

- Customer → Provider: request (informasi ketersediaan)
- Provider → Customer: inform (daftar opsi)
- Customer → Provider: request (pemesanan spesifik)
- Provider → Customer: confirm atau disconfirm (hasil pemesanan)

b. KQML (*contoh skema*)

- Customer → Provider: ask-if (apakah tersedia?) **atau** ask-one (minta satu jawaban).
- Provider → Customer: tell (status/opsi).
- Customer → Provider: achieve (tujuan: lakukan pemesanan).
- Provider → Customer: reply (konfirmasi/penolakan + alasan).

Spesifikasi Domain per Skenario :

Skenario

- **Input Customer:** judul film, tanggal, preferensi jam, kelas kursi, jumlah tiket.

- **Respons Provider:** daftar showtime, kursi tersedia (ID kursi), harga per kursi.
- **Pemesanan:** pilih jam & kursi spesifik → konfirmasi.
- **Error case:** kursi keburu habis saat *checkout* → tawarkan alternatif.

Persyaratan Implementasi

1. **Bahasa/Framework :**
 - Java (JADE)
2. **State Management:**
 - Simpan **log percakapan** (JSON/CSV/SQLite) dengan *timestamp*, pengirim, penerima, performative, conversation-id.
3. **Validasi:**
 - Minimal **2 percobaan** berkasus: satu *happy path*, satu *error path*.
4. **Robustness:**
 - Toleransi *race condition* sederhana (mis. kursi berubah). Boleh pakai *locking* atau *check-again* sebelum konfirmasi.
5. **Dokumentasi Ontologi:**
 - Definisikan skema konten (field wajib/opsional, tipe data, contoh).

Pengumpulan Tugas 1:

1. **Kode sumber** + instruksi menjalankan (README).
2. **Dokumentasi singkat (2–4 halaman)** berisi:
 - Skenario terpilih dan diagram alur (flowchart).
 - Protokol yang dipakai (FIPA-ACL/KQML) dan ontologi pesan.
 - Tangkapan layar/rekaman log *happy path & error path*.
 - Analisis: keputusan desain (state, penanganan error, alternatif).
3. **Log percakapan** dalam format terstruktur (JSON/CSV/format lain).
4. **Link demo video** maksimal 10 menit.

Seluruhnya di .zip dan dikumpulkan di hebat *elearning*.

Rubrik Penilaian (Total 100):

- **Kebenaran Protokol & Performatives (20):** penggunaan FIPA-ACL/KQML tepat, *conversation-id, reply-with/in-reply-to*.
- **Desain Ontologi (15):** field jelas, tipe data konsisten, dapat diperluas.
- **Implementasi Alur (20):** *happy path* lengkap, *error path* tertangani, logging rapi.
- **Manajemen State dan Konsistensi (15):** penanganan stok/slot, *check-again* sebelum konfirmasi.
- **Dokumentasi (10):** ringkas, diagram, contoh pesan.
- **Kualitas Kode (10):** struktur proyek, kejelasan, instruksi jalan.
- **Demonstrasi Video (10)**

Selamat Mengerjakan