

# Tugas Besar Struktur Data

## Semester Ganjil 2025/2026

### LATAR BELAKANG

Mata kuliah **Struktur Data** bertujuan untuk memberikan pemahaman kepada mahasiswa mengenai berbagai jenis struktur data yang biasanya digunakan seperti **Singly Linked List, Doubly Linked List, Stack, Queue, Multi Linked List, Tree dan Graph**. Mahasiswa diharapkan memahami teori yang telah diberikan dan mampu menerapkannya ke kehidupan sehari-hari sehingga dapat digunakan untuk menyelesaikan masalah nyata.

Sebagai salah satu bentuk usaha penerapan teori struktur data yang sudah dipelajari di Mata Kuliah Struktur Data, mahasiswa diminta untuk mencoba **membuat replika program pemutar musik modern**. Layanan pemutar musik modern (seperti Spotify, Apple Music, dll) menuntut pengelolaan koleksi lagu dalam skala besar, pengaturan *playlist* yang fleksibel, antrian pemutaran yang dinamis, pencarian cepat, pengurutan, serta rekomendasi lagu serupa berdasarkan kemiripan data lagu.

### TUJUAN

Tugas besar ini bertujuan untuk :

1. **Melatih mahasiswa** dalam **memilih, merancang dan mengimplementasikan** struktur data yang tepat untuk menyelesaikan suatu studi kasus tertentu
2. **Melatih kemampuan analisis mahasiswa** dalam memilih struktur data yang kompatibel dengan masalah yang dihadapi
3. **Melatih pengalaman praktis** dalam menggunakan alat untuk pengembangan perangkat lunak, seperti Github.

### DESKRIPSI TUGAS BESAR

Bangun **aplikasi konsol Pemutar Musik** yang menyimpan banyak lagu dari beragam artis, genre, album, tahun, dsb. Aplikasi ini dapat **dijalankan oleh** dua peran yaitu **Admin** (mengelola data lagu) **dan User** (memutar dan mengelola playlist). Aplikasi **wajib menggunakan satu atau lebih struktur data yang berbeda** (*singly linked list, doubly linked list, multi linked list, graph, tree, queue, stack*). **Data lagu** disimpan menggunakan tipe data bentukan (*record*).

1. Beberapa **fitur dasar yang wajib** berdasarkan peran pada aplikasi pemutar musik dapat dilihat di bawah ini.
  - a. **Peran Admin**
    - i. Menambah lagu baru ke dalam *library* pemutar musik.

- ii. Melihat semua lagu yang ada di dalam *library* pemutar musik.
- iii. Mengubah data lagu yang ada di dalam *library* pemutar musik.
- iv. Menghapus data lagu yang ada di dalam *library* pemutar musik.

b. **Peran User**

- i. Mencari lagu berdasarkan data lagu (id lagu, judul lagu, dst).
- ii. Memutar lagu dan menghentikan lagu, baik dari *library* aplikasi maupun dari *playlist*.
- iii. Memutar lagu selanjutnya atau sebelumnya (*next* atau *prev* lagu).
- iv. Membuat dan mengatur *playlist* lagu (menambahkan lagu ke *playlist*, menghapus lagu dari *playlist* dan melihat list lagu di *playlist*)

2. **Ketentuan pada fitur dasar** adalah sebagai berikut.

- a. **Data lagu** (id lagu, judul lagu, penyanyi, dll) **ditentukan oleh kesepakatan kelompok** dan dibentuk menggunakan **tipe data record**.
- b. **Memutar dan menghentikan lagu** cukup dengan **perintah untuk play/stop** dan **notifikasi** bahwa lagu tersebut sedang diputar atau lagu tersebut dihentikan.
- c. Jika ada **lagu yang di-update** datanya oleh Admin, maka akan **berubah** juga data lagu yang ada di **playlist User**.
- d. Jika ada **lagu yang dihapus** oleh Admin dari *library*, maka akan **terhapus** juga lagu tersebut dari **playlist User**.
- e. **Next atau prev** lagu :
  - i. Jika User sedang **berada di playlist**, maka **next/prev** akan **mengikuti urutan lagu di playlist**.
  - ii. Jika tidak berada di *playlist* (sedang **berada di halaman utama atau library**), maka **next/prev** akan **memutar lagu yang mirip** dengan lagu sebelumnya.

**Catatan:** Definisi “**mirip**” ditentukan kelompok (contoh: artis sama lebih prioritas daripada *genre* sama). Jika tidak ada lagu yang mirip, tampilkan pesan yang sesuai dan jangan *crash* (boleh *fallback* ke lagu acak).

- f. **Ketentuan** pada fitur dasar **yang tidak diatur**, maka **disesuaikan** dengan kesepakatan **kelompok**.
3. **Aplikasi ini bisa ditambahkan fitur-fitur lain** seperti menjadikan lagu tersebut favorit, membuat daftar favorit, menyusun lagu prioritas, mengurutkan lagu di *library/playlist*, atau lain sebagainya, **baik yang relevan atau bersifat inovatif** dengan pemutaran musik, **menyesuaikan dengan kesepakatan kelompok**.
4. Setiap kelompok **memilih struktur data yang cocok** dan bisa digunakan **untuk menyelesaikan fitur-fitur dasar** yang diminta. Selain itu, setiap kelompok harus **bisa memberikan justifikasi** terhadap pemilihan struktur data tersebut.

5. Boleh **menggunakan operasi *primitive*** dari setiap struktur data **dan sangat disarankan** untuk **membuat *procedure/function* tambahan** untuk mewujudkan fitur yang akan digunakan.
6. **Membuat laporan** yang mencakup :
  - a. Dekripsi **masalah dan fitur** yang diimplementasikan.
  - b. **Penjelasan** mengenai ***procedure/function* yang digunakan atau dibuat** pada aplikasi.
  - c. **Penjelasan** mengenai **struktur data yang dipilih dan digunakan** secara rinci, serta **alasan pemilihan struktur data** tersebut. Disertai dengan **kendala** yang ditemui **dan solusi** yang diambil.
  - d. **Tautan ke *repository* Github** (harus ***public***) yang berisi kode program pemutar musik, dokumentasi aplikasi dan aplikasi yang sudah jadi.
7. **Mengumpulkan laporan** di Learning Management System (LMS).

## PENILAIAN TUGAS BESAR

Beberapa hal yang dinilai dari tugas besar ini adalah sebagai berikut.

1. **Laporan (50%)**, meliputi
  - Kualitas konten laporan tahap 1, 2 dan 3.
  - Kelengkapan dan kerapihan penulisan laporan.
  - Laporan asli yang dibuat oleh kelompok tanpa bantuan AI dan laporan yang sudah dibantu proses *editing*-nya (memperbaiki kejelasan kualitas laporan) oleh AI, **jika menggunakan AI** ketika membuat laporan. Cantumkan perbedaannya.
2. **Implementasi Program (30%)**, meliputi
  - Keberhasilan implementasi fitur-fitur dasar.
  - Efektivitas penggunaan struktur data.
  - **Dokumentasi kode** pada Github yang meliputi ***file README* dan *file Commit Log***.
    - *File README* menjelaskan tentang deksripsi aplikasi, fitur-fiturnya, cara menjalankan program (instruksi instalasi dan penggunaan) dan daftar anggota kelompok.
    - *File Commit Log* berisi cerminan perkembangan program secara bertahap.
3. **Presentasi dan Demo (20%)**, meliputi
  - Kemampuan kelompok mendemonstrasikan dan mempresentasikan hasil kerja kelompok.
  - Kesesuaian program yang dikerjakan dengan laporan yang diberikan.

## TIMELINE TUBES

Berikut adalah timeline dari pengerjaan tubes ini.

1. **Laporan tahap 1 (bobot 10% dari nilai Laporan)** berisi latar belakang, analisis masalah, dan perencanaan solusi. **Dikumpulkan pada pekan ke 11** (30 November 2025) melalui LMS **setelah melakukan pendampingan tubes** bersama asprak. Sertakan bukti hasil pendampingan tubes berupa hasil diskusi.
2. **Laporan tahap 2 (bobot 15% dari nilai Laporan)** berisi fitur dan struktur data yang sudah diimplementasikan, kendala dan solusi yang ditemukan, dan fitur-fitur yang akan dibuat. **Dikumpulkan di pekan ke 12** (7 Desember 2025) melalui LMS.
3. **Laporan tahap 3 (bobot 25% dari nilai Laporan)** berisi pendahuluan dari laporan tahap 1, deskripsi proyek yang dikembangkan beserta dengan semua fitur dan struktur data yang digunakan, tantangan dan solusi yang dihadapi ketika mengembangkan aplikasi, kesimpulan dan rekomendasi, link Github dan referensi yang digunakan selama pengerjaan tugas. **Dikumpulkan pada pekan ke 13** (14 Desember) melalui LMS.
4. **Presentasi dan demo tubes** dimulai dari pekan ke 14 sampai pekan ke 16.
5. **Teknis pelaksanaan presentasi dan demo tubes** diserahkan kepada masing-masing dosen pengampu.