

Progres 1, 2 & 3 Tugas Besar Struktur Data Sistem Manajemen Rute Bus dan Halte



Program Studi Teknologi Informasi

Universitas Telkom Surabaya

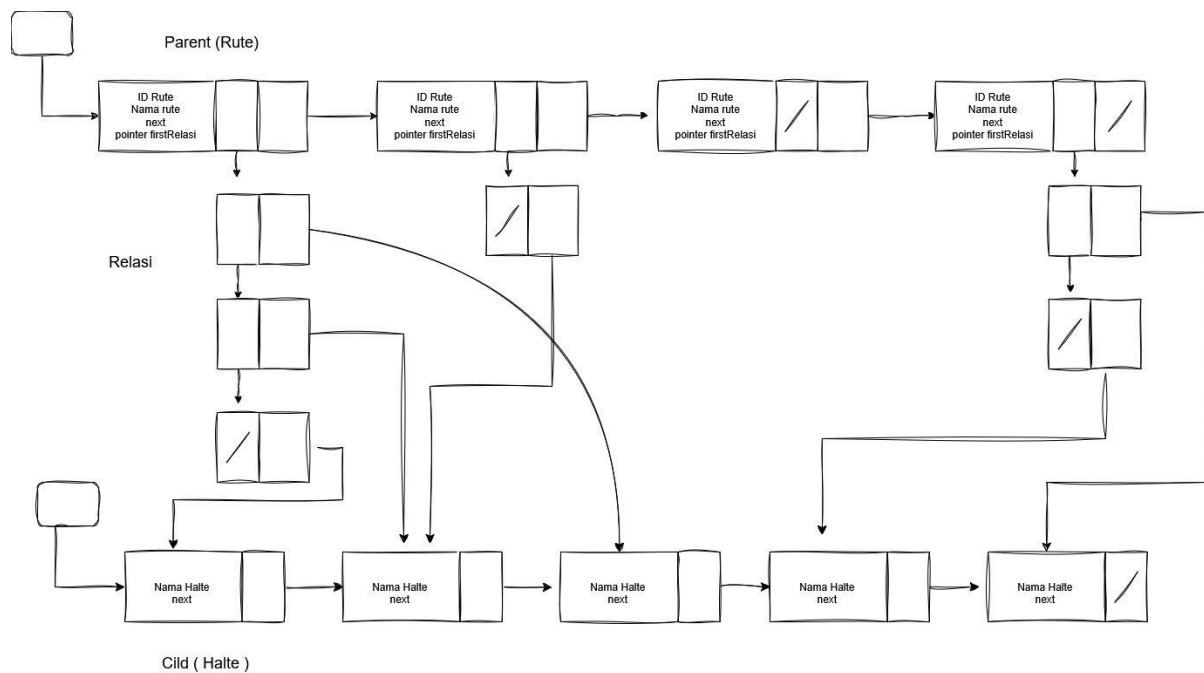
Oleh :

M. Ghymvandyar El M (103092400087)

LAURA DJASMIN TANDIAWAN (103092400077)

Deskripsi Kasus Detail

Sistem Transportasi bus kota sangat membutuhkan pengelolaan data Rute dan Halte yang terstruktur. Jika data ini dicatat secara manual, pembaruan rute tentu menjadi sulit, hubungan antar halte tidak jelas, dan pencarian informasi seperti “halte apa saja yang dilalui rute X” atau rute apa saja yang melewati halte Y” menjadi lambat, karena itu diperlukan model data yang fleksibel, mudah diperbarui, dan mampu menaganani hubungan banyak - ke - banyak.



Kelompok kami merancang **Sistem Manajemen Rute Bus dan Halte** menggunakan Mlti Linked List Tipe A untuk memodelkan hubungan(many to many) antara Rute dan Halte dengan struktur:

1. List Parent = Rute

Berisi data setiap Rute bus, seperti:

- ID Rute
- Nama Rute
- pointer next (lokasi)
- pointer firstRelasi

Setiap node rute memiliki pointer firstRelasi yang menunjuk ke daftar halte yang dilalui rute tersebut.

2. List Child = Halte

Berisi daftar setiap halte bus, seperti:

- Nama Halte
- next

Node halte tidak menyimpan pointer relasi secara langsung, melainkan dihubungkan melalui list relasi yang dimiliki oleh node rute.

3. List Relasi = Penghubung Rute ↔ Halte

List Relasi berfungsi sebagai penghubung antara **Rute (Parent)** dan **Halte (Child)**.

Setiap node relasi menyimpan alamat sebuah node Halte dan ditempatkan pada sisi Rute sebagai parent. Dengan demikian, satu rute dapat memiliki banyak halte, dan satu halte dapat dilalui oleh banyak rute.

Struktur ini memodelkan hubungan:

- Satu rute → banyak halte
- Satu halte → banyak rute

PENJELASAN SISTEM

Sistem Manajemen Rute Bus mampu mengelola data rute, data halte, serta hubungan banyak-ke-banyak di antara keduanya secara terstruktur menggunakan Multi Linked List Tipe A. Setiap rute disimpan sebagai node parent yang memiliki daftar relasi ke halte-halte yang dilaluinya, sehingga hubungan antar data dapat dikelola secara dinamis dan efisien.

Penempatan relasi pada sisi **Rute sebagai parent** memungkinkan proses pencarian dan pengelolaan data menjadi lebih cepat, misalnya ketika operator ingin mengetahui daftar halte yang dilalui oleh suatu rute tertentu atau menambahkan rute baru pada jaringan yang sudah ada tanpa mengganggu data lain.

Dalam kondisi nyata, kebutuhan seperti pembaruan jalur bus, penataan ulang halte, maupun pengecekan rute alternatif dapat dilakukan dengan mudah tanpa harus mengubah struktur data secara keseluruhan, sehingga sistem tetap fleksibel dan mudah dikembangkan.

SPESIFIKASI PROGRAM TIPE A

Sistem yang dibangun harus mengimplementasikan Multi Linked List Tipe A untuk memodelkan hubungan banyak-ke-banyak antara Halte (Parent) dan Rute (Child).

- insertData Rute(parent)
- insert data Halte(child)
- Insert Relasi Rute ↔ Halte
- Delete Rute
- Delete Halte
- Delete Relasi
- Find Rute
- Find Halte
- Find Relasi
- ShowAllRute
- ShowAllHalte
- showChildFromParent (fitur I)

- showParentChildList (fitur m)
- showChildParentList (fitur n)
- showParentFromChild (fitur o)
- countRelationParent (fitur p)
- countRelationChild (fitur q)
- countChildNoRelation (fitur r)
- editRelation (fitur s)

Deskripsi Requirement:

- a. Insert element parent (Rute) : 5
PIC : Laura

Nama Fungsi : InsertRute

Penjelasan : *Menambah data rute yang baru ke dalam List Parent (Rute).*

- b. Insert element child (Halte) : 5

PIC : Vandyar

Nama Fungsi : InsertHalte

Penjelasan : *Menambahkan data halte baru ke dalam List Child (Halte)*

- c. Insert element relation : 5

PIC : Laura

Nama Fungsi : InsertRelasi

Penjelasan : Membuat hubungan antara layanan Halte dengan Rute

- d. Delete element parent (Rute) : 5

PIC : Laura

Nama Fungsi: deleteRute

Penjelasan: Menghapus data rute dari List Parent beserta semua relasi yang terhubung

- e. Delete element child (HALTE) : 5

PIC : Vandyar

Nama Fungsi : DeleteHalte

Penjelasan : Menghapus data layanan child dari List Child serta menghapus relasi yang berhubungan

f. Delete element relation : 5

PIC : Bersama

Nama Fungsi : DeleteRelasi

Penjelasan : Menghapus salah satu dari layanan tertentu

g. Find element Parent (Rute) : 5

PIC: Laura

Nama Fungsi: FindParent

Penjelasan: Mencari data rute berdasar ID rute.

h. Find element child (Halte) : 5

PIC : Vandyar

Nama Fungsi : FindHalte

Penjelasan : Mencari data layanan berdasarkan nama Halte

i. Find element relation : 5

PIC : Laura

Nama Fungsi : FindRelasi (findRelasi)

Penjelasan : Mengecek apakah suatu Halte terhubung Rute tertentu

j. Show all data di List Parent (Rute) : 5

PIC: vandyar

Nama Fungsi: showAllParent (showAllRute)

Penjelasan: Menampilkan seluruh rute yang ada di sistem

k. Show all data di List Child (Halte) : 5

PIC : Vandyar

Nama Fungsi : PrintAllChild(showAllHalte)

Penjelasan : Menampilkan seluruh halte yang ada dalam sistem

- l. Show data child dari parent tertentu : 5

PIC : Bersama

Nama Fungsi : PrinttChildFromParent (showChildFromParent)

Penjelasan : Menampilkan daftar halte yang dilalui oleh suatu rute tertentu.

- m. Show setiap data parent beserta data child yang berelasi dengannya : 5

PIC : Bersama

Nama Fungsi : PrintParentFromChild(showAllParentWithChild)

Penjelasan: Menampilkan seluruh rute beserta halte yang berelasi dengannya.

- n. Show data child beserta data parent yang masing-masing child miliki : 10

PIC : Vandyar

Nama Fungsi : PrintAllRelasiFromParent (showAllData)

Penjelasan : Menampilkan seluruh Rute beserta Halte yang dilaluinya dan halte yang tidak memiliki relasi.

- o. Show data parent yang berelasi dengan child tertentu (Rute) : 5

PIC : vandyar

Nama Fungsi: PrintAllRelasiFromChild (showParentOfChild)

Penjelasan: Menampilkan rute-rute yang melewati halte tertentu.

- p. Count relation dari setiap element parent : 5

PIC : Laura

Nama Fungsi : CountChildFromParent (countRelasiParent)

Penjelasan : Menghitung jumlah halte yang dimiliki oleh suatu rute.

- q. Count relation yang dimiliki oleh child tertentu : 5

PIC : Vandyar

Nama Fungsi : CountParentFromChild (countRelasiChild)

Penjelasan : Menghitung jumlah rute yang melewati halte tertentu.

- r. Count element child yang tidak memiliki relasi : 5

PIC : Vandyar

Nama Fungsi : CountFreeChild (countChildNoRelasi)

Penjelasan : Menghitung jumlah halte yang tidak terhubung dengan rute manapun.

- s. Edit relasi /mengganti child dari parent tertentu : 5

PIC : Vandyar

Nama Fungsi : EditRelasi (editRelasi)

Penjelasan : *Proses dilakukan dengan menghapus relasi lama kemudian menambahkan relasi baru agar struktur relasi tetap konsisten.*

1. presentasi kontribusi anggota

vandyar 50%

- Membuat Sturct halte
- Implementasi insert/delete halte
- Find halte & find relasi
- Laporan bagian parent dan relasi
- Show kompleks
- Implemntasi main.cpp
- perancangan menu dan CLI
- pengujian program dan simulasi responsi

laura 50%

- Membuat Sturct Rute
- Implementasi insert/delete rute
- Insert & delete relasi
- Show rute, count rute tanpa relasi
- Laporan bagian child dan pengujian

LAPORAN Progress 1 (40%)
BERSAMA MAS KADEK GANDHI WAHYUDI
TANGGAL 03 desember 2025

dokumentasi :



LAPORAN Progress 2 (80%)
BERSAMA MAS MICHEL ANGGELO
TANGGAL 10 desember 2025

dokumentasi :



LAPORAN Progress 3 (100%)
BERSAMA MAS KADEK GANDHI WAHYUDI
TANGGAL 17 desember 2025

dokumentasi :



CLI yang di minta asprak

menu utama dan antar muka CLI

PIC : Vandyar

Nama fungsi : main

penjelasan :

Mengelola alur program melalui menu berbasis Commmand Line Interface (CLI)
yang menghubungkan input pengguna dengan seluruh fungsi pengolahan data : Rute, R
Halte dan Relasi.

MANAJEMEN RUTE DAN BUS

===== SISTEM MANAJEMEN RUTE BUS =====

1. Tambah Rute
2. Tambah Halte
3. Tambah Relasi Rute - Halte
4. Tampilkan Semua Rute
5. Tampilkan Semua Halte
6. Tampilkan Halte dari Rute
7. Tampilkan Rute dari Halte
8. Edit Relasi
9. Hapus Rute
10. Hapus Halte
11. Tampilkan Semua Data
0. Keluar

Pilih: |