

UNGUIDED DAN TUGAS

MODUL 11 – RECORD

Tipe D

UNGUIDED:

Atma Hospitals merupakan sebuah rumah sakit yang dimiliki Universitas Atma Raya. Saat ini, Atma Hospitals memerlukan bantuan Anda untuk menyimpan data pasien yang dirawat. Setelah melakukan wawancara bersama manajemen, berikut beberapa poin penting yg dicatat:

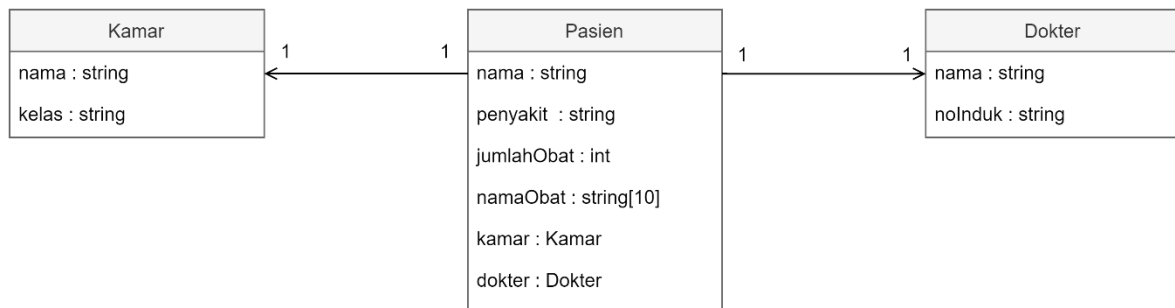
Untuk sekarang, cukup 1 data pasien saja yang dicatat ya [mas/mbak], sekadar proof-of-concept. ... Oya, tidak perlu ada login form juga, asumsinya siapapun yang masuk tuh udah dijamin terotorisasi. Nanti saya kasih dokumen spesifikasi programnya.

Setelah wawancara tersebut, Anda diberikan dokumen spesifikasi program sebagai berikut:

ATMA HOSPITALS SYSTEM SPECIFICATION

Last updated: 24/05/2022

Diagram kelas:



Sistem wajib mencatat nama pasien, penyakit yang diderita, obat yang diterima, dan dokter yang memeriksa. Data kamar opsional.

1. Menu 1: Input Data Perawatan

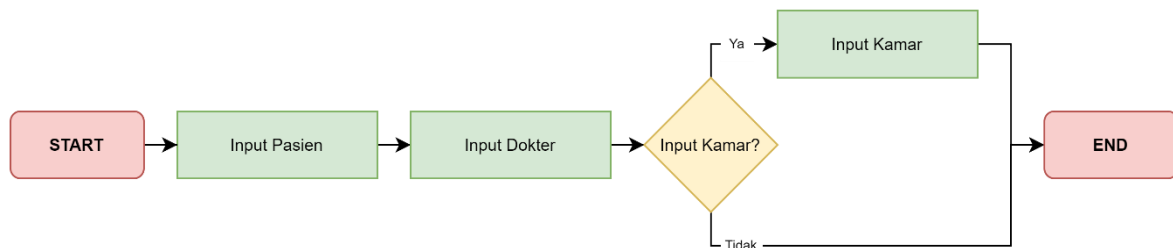
Menu ini hanya dapat diakses apabila data pasien belum diisi.

User dapat menginput data pasien yakni:

- Nama pasien
- Penyakit yang diderita
- Jumlah obat yang diberikan (min. 1, maks. 10 → terapkan error handling)
- Daftar nama obat yang diberi (diinput sebanyak jumlah obat yang diberikan)

Setelah menginput data pasien, user diwajibkan mengisi data dokter. Setelah itu, user diberikan pilihan untuk menginput data kamar. Apabila IYA, tampilkan form input data kamar.

Apabila ditulis dalam diagram alir, berikut alurnya:



Setelah selesai menginput data, kembalilah ke menu utama.

Error handling: nama pasien tidak boleh kosong, nama dokter tidak boleh kosong, nama kamar (apabila diinput) tidak boleh kosong.

2. Menu 2: Tampilkan Data Perawatan

Menu ini hanya dapat diakses apabila data pasien sudah diisi.

Pada menu ini, tampilkanlah inputan data yang telah diisi dengan contoh format/template sebagai berikut:

```
[Pasien]
Nama Pasien : <nama>
Penyakit    : <penyakit>
[Jumlah Obat: <jumlahObat>]
1. <namaObat[0]>
2. ...

[Dokter]
Nama        : <dokter.nama>
No. Pegawai : <dokter.noInduk>

{{IF: <kamar> is not empty:}}
[Kamar]
Nama        : <kamar.nama>
Kelas Kamar : <kamar.kelas>

{{ENDIF}}
```

Penjelasan:

Tampilkanlah data pasien sebagaimana mestinya, sesuai dengan *field* yang ada, dilanjutkan dengan data dokter yang memeriksa, kemudian tampilkan data kamar jika sudah diisi.

3. Menu 3: Ubah Data

Menu ini hanya dapat diakses apabila data pasien sudah diisi.

Di dalam menu ini, tambahkan dua buah submenu dengan ketentuan:

1) Submenu 1: mengubah data pasien

Submenu ini berfungsi untuk mengubah data pasien saja, **tidak termasuk obat yang diberikan dan dokter/kamarnya.**

2) Submenu 2: mengubah data dokter

Submenu ini berfungsi untuk mengubah data dokter saja.

3) Submenu 3: mengubah data kamar

Submenu ini berfungsi untuk mengubah data kamar. Apabila belum pernah diinput, menu ini berfungsi untuk menginput data kamar.

Setelah mengisi salah satu data di atas, kembalilah ke menu utama.

Error handling: nama pasien tidak boleh kosong, nama dokter tidak boleh kosong, nama kamar tidak boleh kosong.

4. Menu 4: Hapus Data

Menu ini hanya dapat diakses apabila data pasien sudah diisi.

Sebelum menghapus, tambahkanlah konfirmasi apakah user benar-benar ingin menghapus data. Apabila iya, kosongkanlah data pasien, dokter, dan kamar.

5. Menu 0: Keluar Program

Menu ini dapat diakses entah data pasien sudah diisi maupun belum diisi.

Sebagai ucapan terima kasih, silakan cantumkan nama developer di sini.

Format: “<Nama> – <NPM lengkap> – Praktikum Dasar Pemrograman <Kelas>”

Contoh identitas: “Jolly Hans Frankle – 200710932 – Praktikum Dasar Pemrograman D”

BONUS

1. Menambahkan error handling pada field di struct Pasien: (+6)

- Nama obat yang diinput tidak boleh berupa string kosong.
- Nama obat harus unik (tidak boleh memiliki nama sama).
- Penyakit tidak boleh kosong.

2. Menambahkan error handling pada field di struct Dokter: (+1)

- Nomor induk harus berisi 9 karakter.

3. Menambahkan error handling pada field di struct Kamar: (+1)

- Kelas kamar antara Kelas I, Kelas II, Kelas III, VIP, atau VVIP.

4. Pada menu 3 (Ubah Data), tambahkan submenu 4: Ganti semua obat (+5)

[Bonus ini akan diperiksa jika bonus 123 sudah diimplementasikan dengan benar.](#)

Menu ini berfungsi untuk mengganti semua obat yang ada dalam daftar, sehingga akan meminta inputan jumlah obat yang diberikan, kemudian meminta inputan nama obat sebanyak jumlah obat yang diberikan, sama halnya saat hendak menginput daftar obat di menu input data (menu 1).

5. Pada menu 3 (Ubah Data), tambahkan submenu 5: Ubah salah satu obat (+7)

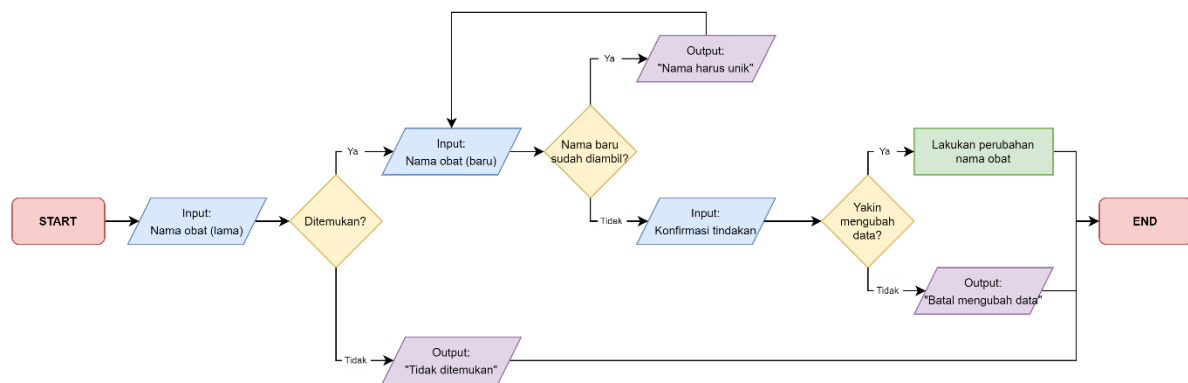
[Bonus ini akan diperiksa jika bonus 1234 sudah diimplementasikan dengan benar.](#)

Guna mengantisipasi kesalahan input nama obat, tambahkanlah menu ubah nama obat. Menu ini berfungsi untuk mengubah salah satu nama obat, dicari berdasarkan nama yang lama.

Saat memasuki menu ini, user akan memasukkan nama obat yang ingin diubah. Apabila ditemukan, mintalah inputan untuk nama baru obat tersebut, jangan lupa cek apakah obat tersebut sudah ada di daftar atau belum (ingat, daftar nama obat harus unik). Setelah input, tambahkan konfirmasi apakah user yakin ingin mengganti nama obat. Apabila IYA, gantilah nama obat tersebut sesuai nama baru yang diinput user.

Diagram alir ada di halaman selanjutnya...

Berikut alur proses perubahan nama obat:



KETENTUAN BONUS

1. Tidak ada pembatasan bertanya kepada asisten. Pemahaman itu lebih penting.
2. Bonus 1, 2, dan 3 (error handling) dapat diklaim walaupun UGD belum selesai.
3. Bonus 4 dan 5 hanya dapat diklaim apabila bonus 1, 2, dan 3 sudah diimplementasikan dan berjalan semestinya.
4. Apabila telah yakin bonus 12345 telah dikerjakan dengan benar, jangan lupa mengupload file yang sama juga ke uploader tugas **sebelum cut-off** (15 menit setelah waktu praktikum berakhir).
5. Nilai maksimum setelah mengerjakan bonus adalah **UGD 120, TGS 100**, bobot masing-masing bonus bisa dilihat di atas.
6. **Ada format pengumpulan khusus bonus, lihat di bawah.**

TUGAS

Menyelesaikan UGD dan bonus 1–5.

KETENTUAN Pengerjaan

1. **Tambahkan 5 digit terakhir NPM di belakang setiap struct.** Cth: MataKuliah_10932.
Konsekuensi melanggar: UGD/TGS -5
2. Project harus memiliki struktur *record/struct* sesuai dengan diagram kelas yang disertakan (di halaman pertama). Nama *field* di dalam record tersebut tidak harus sama.
Konsekuensi melanggar: UGD/TGS -20
3. Dikerjakan dalam proyek Dev-C++, menggunakan bahasa pemrograman C.
Konsekuensi melanggar: UGD/TGS diberi nilai 50% dari semestinya
4. Wajib menyertakan identitas praktikan saat keluar program.
Konsekuensi melanggar: UGD/TGS -10

5. Tidak diperbolehkan menggunakan *array of record*, sebab tidak diperlukan juga *array of record* dalam pengerjaan UGD maupun tugas.

Konsekuensi melanggar: UGD/TGS diberi nilai 50% dari semestinya

6. 40 menit pertama belum diperbolehkan bertanya kepada asisten.

KETENTUAN PENGUMPULAN

Unguided:

Non-bonus: **UGD11_X_YYYYY**

Bonus: **UGD11_BONUS_Z_X_YYYYY**

Tugas:

Non-bonus: **TGS11_X_YYYYY**

Bonus: **TGS11_BONUS_X_YYYYY** (*tidak perlu menyertakan nomor bonus (Z)*)

Keterangan:

X = kelas

Y = 5 digit terakhir NPM

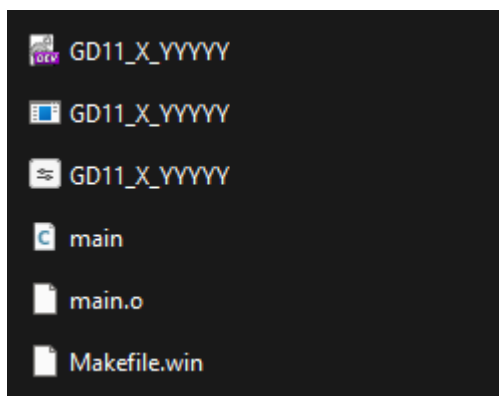
Z = nomor bonus

Contoh:

Non-bonus: **UGD11_D_10932**

Bonus: **UGD11_BONUS_1234_D_10932** (*berarti hendak klaim bonus 1, 2, 3, dan 4*)

Dikumpulkan secara lengkap di dalam satu folder, **kemudian di-ZIP (bukan RAR)**:



Format pengumpulan tidak sesuai: UGD/TGS -10