

LAPORAN TUGAS BESAR PENGENALAN KOMPUTASI
KU1102
HASIL EKSPLORASI CARA KERJA AUTOMATIC TELLER MACHINE (ATM)



DISUSUN OLEH
KELOMPOK 8 - KELAS 23

GAGAS PRAHARSA BAHAR	16520289
MALIK AKBAR HASHEMI R.	16520299
ALIFIA RAHMAH	16520309
NG KYLE	16520319

DAFTAR ISI

I. LAPORAN INTI	3
1. 1 Tugas 1	3
1.1.1 Eksplorasi ATM	3
1.1.2 Dekomposisi	3
1.2 Tugas 2	4
1.2.1 Deskripsi Simulasi	4
1.2.1.1 Input	4
1.2.1.2 Proses	5
1.2.1.3 Output	5
1.2.2 Algoritma	7
1.2.2.1 Flowchart	7
1.2.2.1.1 Main Flowchart	7
1.2.2.1.2 Sub-Program	7
1.2.2.2 Pseudocode	11
1.3 Tugas 3	15
1.3.1 Pendekatan	15
1.3.2 Source code program	15
1.3.3 Screenshot Program	15
II. KESIMPULAN DAN LESSON LEARNED	16
2.1 Kesimpulan	16
2.2 Lesson Learned	16
III. PEMBAGIAN TUGAS DALAM KELOMPOK	16
IV. DAFTAR REFERENSI	16

I. LAPORAN INTI

1. 1 Tugas 1

1.1.1 Eksplorasi ATM

ATM didefinisikan KBBI : “Anjungan Tunai Mandiri (ATM) mesin otomatis (milik bank) untuk mengeluarkan uang tunai dengan teknik tertentu, seperti menekan tombol nomor tabungan, menekan nomor (kode) sesuai dengan petunjuk”

ATM Processing adalah sekumpulan langkah yang membuat sehingga sebuah mesin ATM dapat menampilkan informasi akun user ATM. Setelah menerima PIN, mesin akan berkomunikasi dengan bank pengguna dan mengakses informasi di database mengenai informasi pemilik kartu. (<https://americanatm.com/atm-processing/>)

Diketahui bahwa ATM setidaknya memiliki berapa fungsi:

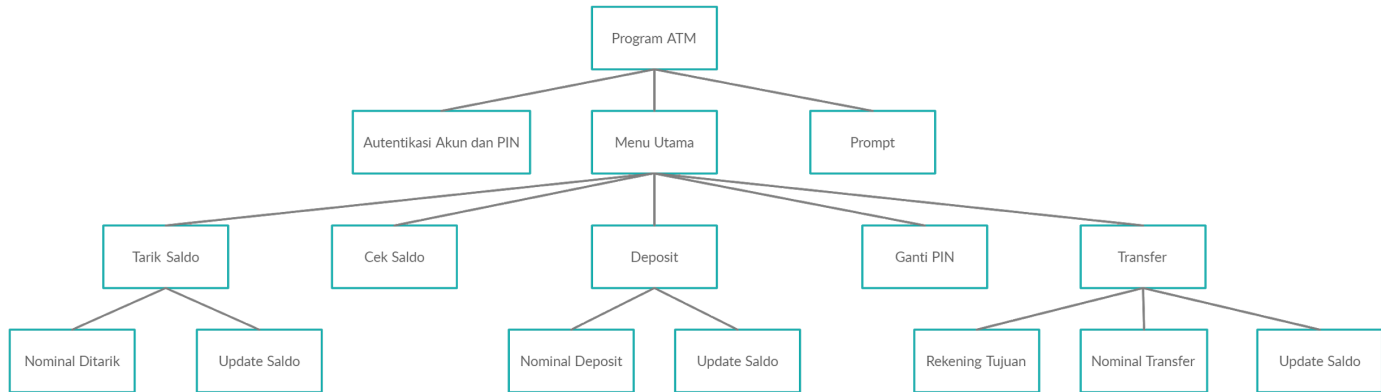
1. Menerima PIN dan mendisplay informasi akun.
2. Melakukan transaksi (transfer, cek saldo, menarik saldo).
3. Fungsi opsional (Mengganti PIN dan Deposit Uang).

1.1.2 Dekomposisi

Dari fungsi-fungsi umum dan opsional dari mesin ATM kita dapat mendekomposisi mesin ATM sebagai berikut :

1. Autentikasi Rekening dan PIN
2. Tarik Saldo, Cek Saldo, Deposit, Transfer, Ganti PIN. (Menu Utama)
 - a. Tarik Saldo
 - i. Nominal Ditarik
 - ii. Update Saldo
 - b. Cek Saldo
 - c. Deposit
 - i. Nominal Deposit
 - ii. Update Saldo
 - d. Transfer
 - i. Rekening tujuan
 - ii. Nominal
 - iii. Update Saldo (Transfer dan ditransfer)
 - e. Ganti PIN
 - i. Memasukkan PIN awal
 - ii. Memasukkan PIN baru

3. Prompt (Setiap Sub-Menu)



Gambar 1.1 Dekomposisi

1.2 Tugas 2

1.2.1 Deskripsi Simulasi

1.2.1.1 Input

Input dilakukan pada beberapa bagian program :

1. Autentikasi Akun dan PIN
Input nomor rekening dan PIN. Keduanya harus numerik. (Untuk PIN numerik dengan panjang 6). Input [y/n] bila salah menginput nomor rekening.
2. Menu Utama (Menu 0)
Input angka sub-menu. Pada program ini terdapat 5 sub-menu, sehingga nilai input yang sesuai adalah angka 1-5
3. Sub-menu Tarik Saldo (Menu 2)
Input nominal saldo yang akan ditarik.
4. Sub-menu Deposit (Menu 3)
Input nominal yang akan dideposit.
5. Sub-menu Transfer (Menu 4)
Terdapat 2 field input, field 1 menginput Nomor Rekening tujuan, untuk field 2 menginput nominal yang ditransfer.
6. Sub-menu Ganti PIN (Menu 5)
Input PIN pengganti (dengan syarat PIN harus numerik panjang 6)

7. Prompt

Input [y/n]

Keterangan :

- Setiap nominal hanya perlu diisi nilainya saja (contoh : 150000).
- y = yes / ya
- n = no / tidak

1.2.1.2 Proses

Proses dilakukan pada beberapa bagian program :

1. Autentikasi PIN
Memeriksa daftar PIN di tabel data rekening, bila benar akan lanjut ke menu utama, jika gagal 3 kali akan mengakibatkan pemblokiran akun.
2. Sub menu Tarik Saldo (Menu 2)
Mengurangi saldo rekening pengguna sebesar input, dan menyimpan jumlah saldo di tabel data rekening.
3. Sub menu Deposit Tunai (Menu 3)
Menambahkan saldo rekening pengguna sebesar input, dan menyimpan jumlah saldo di tabel data rekening.
4. Sub menu Transfer Rekening (Menu 4)
Mengurangi saldo rekening pengguna sebesar input, menambahkan saldo rekening lain yang dipilih pengguna sebesar input, dan menyimpan jumlah saldo di tabel data rekening.
5. Sub menu Ganti PIN (Menu 5)
Mengubah PIN di tabel data rekening sesuai input.
6. Prompt kembali ke sub menu atau keluar dari program

1.2.1.3 Output

Output dilakukan pada beberapa bagian program :

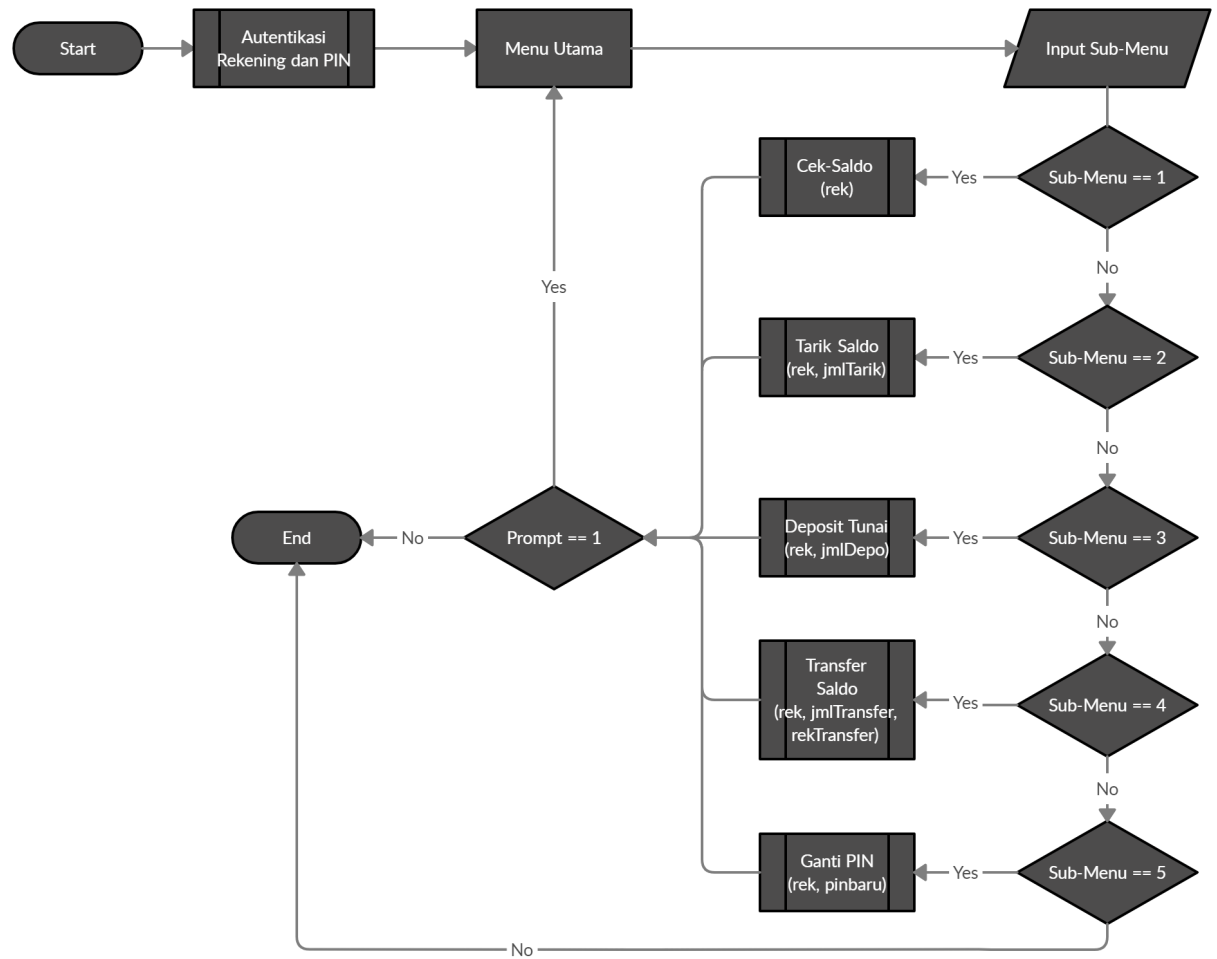
1. Autentifikasi nomor rekening dan PIN serta prompt [y]
Mencetak tulisan “ATM Pengkom” dengan antarmuka *command-line interface* dan juga lima pilihan menu.
2. Sub menu Cek Saldo (Menu 1)
Mencetak sisa saldo.
3. Sub menu Tarik Saldo (Menu 2)
Mencetak pernyataan bahwa berhasil menarik saldo dan mencetak sisa saldo.

4. Sub menu Deposit Tunai (Menu 3)
Mencetak pernyataan bahwa berhasil mendeposit uang dan mencetak sisa saldo.
5. Sub menu Transfer Saldo (Menu 4)
Mencetak pernyataan bahwa berhasil mentransfer saldo dan mencetak sisa saldo.
6. Sub menu Ganti PIN (Menu 5)
Mencetak pernyataan bahwa berhasil mengganti PIN.
7. Prompt [n]
Mencetak pernyataan terima kasih telah menggunakan ATM Pengkom.

1.2.2 Algoritma

1.2.2.1 Flowchart

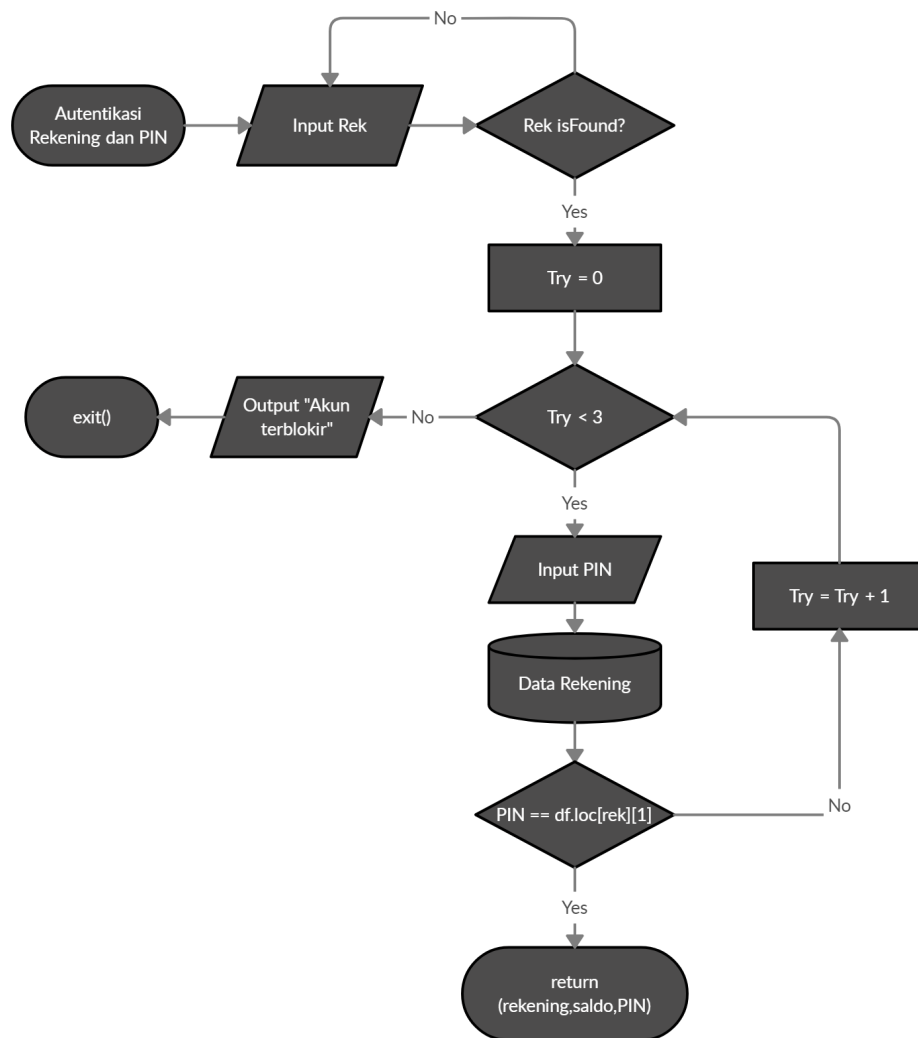
1.2.2.1.1 Main Flowchart



Gambar 1.2 Main Flowchart

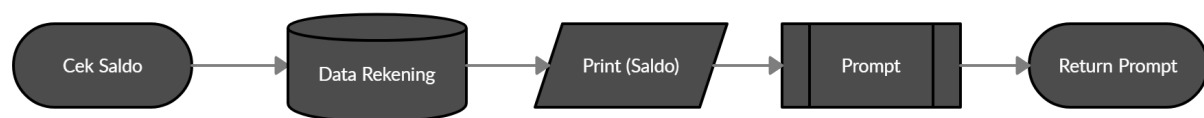
1.2.2.1.2 Sub-Program

1. Autentikasi Rekening dan PIN



Gambar 1.3 Flowchart Autentikasi Rekening dan PIN

2. Cek Saldo



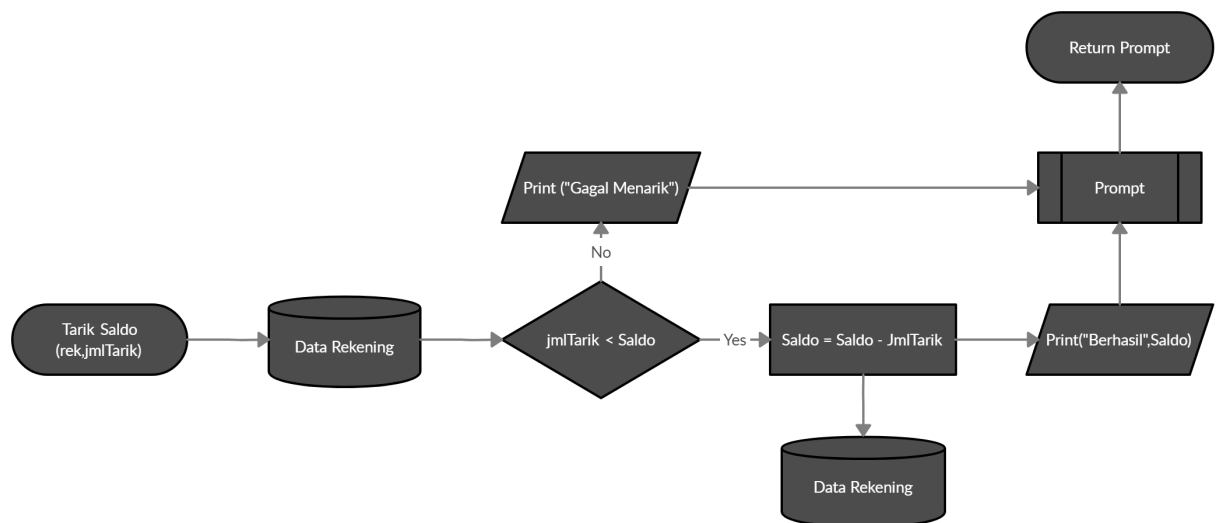
Gambar 1.4 Flowchart Cek Saldo

3. Transfer Saldo



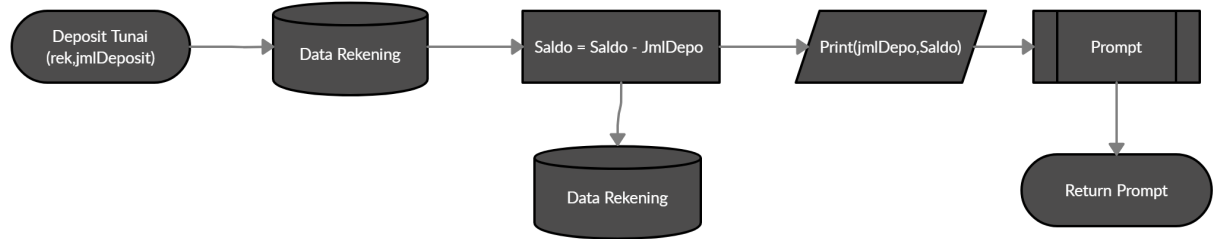
Gambar 1.5 Flowchart Transfer Saldo

4. Tarik Saldo



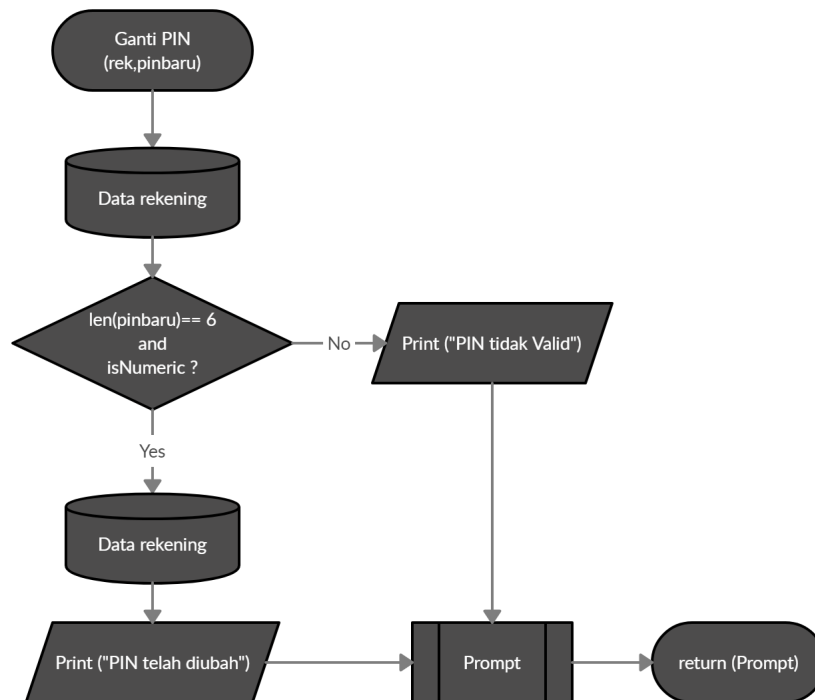
Gambar 1.6 Flowchart Tarik Saldo

5. Deposit Tunai



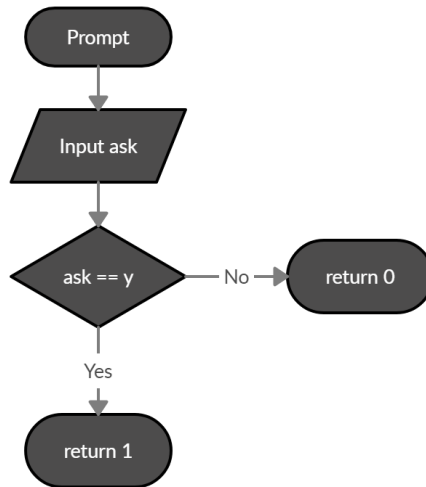
Gambar 1.7 Flowchart Deposit Tunai

6. Ganti PIN



Gambar 1.8 Flowchart Ganti PIN

7. Prompt



Gambar 1.9 Flowchart Prompt

1.2.2.2 Pseudocode

```
PROGRAM SimulasiATM
{ Deskripsi: Program simulasi cara kerja ATM }

KAMUS
{ df : array[3][3] of integer }
{ rek, pin, saldo, tries, me, jumlahTarik, jumlahDepo,
rekTransfer, pinbaru : integer}

ALGORITMA
{ Kumpulan fungsi dan prosedur }

function menu() int
{ Mencetak menu ke layar dan menerima input pilihan
dari pengguna }
KAMUS LOKAL
{ menu : integer }
ALGORITMA
output ("-----")
output ("ATM PENGKOM")
output ("-----")
output ("Pilih menu:")
output ("1. Cek Saldo")
output ("2. Tarik Saldo ")
output ("3. Deposit Tunai")
output ("4. Transfer Saldo")
output ("5. Ganti PIN ")
input(menu)
return menu

function prompt() bool
{ Memberi pilihan kepada pengguna apakah akan
melanjutkan pemakaian atau tidak }
```

```

KAMUS LOKAL
{ ask : string }
ALGORITMA
input(ask)
if (ask == 'y') then
    return 1
else
    output("Terima kasih telah menggunakan ATM
Pengkompom.")
    return 0
endif

procedure cekSaldo(rek: integer)
{ Mencetak saldo rekening pengguna }
KAMUS LOKAL
{ rek, saldo : integer }
ALGORITMA
saldo  $\square$  df[rek][1]
output("Saldo anda sekarang Rp" + saldo + ".")

procedure tarikSaldo(rek, jumlahTarik: integer)
{ Mengurangi saldo pengguna sesuai jumlah yang ingin
ditarik dan mencetak sisa saldo }
KAMUS LOKAL
{ rek, jumlahTarik, saldo : integer}
ALGORITMA
saldo  $\square$  df[rek][1]
saldo  $\square$  saldo - jumlahTarik
if (saldo < 0) then
    print("Gagal menarik uang. Jumlah uang yang
ingin ditarik lebih besar dari saldo.")
else
    print("Berhasil menarik uang sebanyak Rp" +
jumlahTarik + ".")
    print("Sisa saldo anda sekarang Rp" + saldo +
".")
    df[rek][1]  $\square$  saldo
    writefile('data.csv')
endif

procedure deposit(rek, jumlahDepo)
{ Menambah saldo pengguna sesuai jumlah yang ingin
dimasukkan}
KAMUS LOKAL
{ rek, jumlahDepo, saldo : integer }
ALGORITMA
saldo  $\square$  df[rek][1]
saldo  $\square$  saldo + jumlahDepo
output("Berhasil mendeposit uang sebanyak Rp" +
jumlahDepo + ".")
output("Sisa saldo anda sekarang Rp" + saldo + ".")
df[rek][1]  $\square$  saldo
writefile('data.csv')

procedure transfer(rek, jumlahTransfer, rekTransfer :
integer)
{ Mengurangi saldo pengguna, dan menambah saldo di
rekening lain yang dipilih pengguna sesuai jumlah
yang ingin ditransfer }
KAMUS LOKAL
{ rek, jumlahTransfer, rekTransfer, saldoa, saldor :

```

```

integer }
ALGORITMA
try
    saldor = df[rekTransfer][1]
except
    output("Rekening yang anda masukkan tidak valid.")
    saldo = df[rek][1]
    saldo = saldo - jumlahTransfer
    if (saldo < 0) then
        output("Gagal transfer uang. Jumlah uang yang ingin ditransfer lebih besar dari saldo.")
        saldo = saldo + jumlahTransfer
    else
        output("Berhasil transfer uang sebanyak Rp" + jumlahTransfer + ".")
        output("Sisa saldo anda sekarang Rp" + saldo + ".")
        saldor = saldor + jumlahTransfer
        df[rek][1] = saldo
        df[rekTransfer][1] = saldor
        writefile('data.csv')
    endif

procedure gantiPIN(rek, pinbaru: integer)
{ Mengganti pin ATM pengguna untuk autentikasi berikutnya }
KAMUS LOKAL
{ rek, pinbaru : integer }
ALGORITMA
if (len(pinbaru) == 6) and (pinbaru.isnumeric()) then
    df[rek][0] = pinbaru
    print("Berhasil mengubah PIN.")
    writefile('data.csv')
    else
        output("PIN baru yang anda masukkan tidak valid.")
    endif

{ inisialisasi dataframe }
df = readfile("data.csv")

tries = 0
rek = ""

{ Program utama }

{ Penyocokan rekening dengan yang ada di database }
while (rek not in df["PIN"]) do
    input(rek)
    if (rek not in df["PIN"]) then
        output("Rekening anda tidak valid. Silahkan coba lagi.")
    endif
endwhile

{ Autentikasi PIN }
while tries < 3 do
    try
        input(pin)
        if (pin == df[rek][0]) then

```

```

        saldo = df[rek][1]
        break
    else { pin != df[rek][0] }
        raise KeyError
    endif
except KeyError
    if (tries < 2) then
        output("PIN anda salah. Coba lagi. (" +
            (2 - tries) + " percobaan lagi).")
    else
        output("Kartu ATM anda terblokir.
Silahkan hubungi Bank terdekat.")
        exit()
    endif
    tries = tries + 1
endwhile

{ Main menu }
me = 0
while True do
    me = menu()
    if (me == 1) then
        { Pilihan menu pertama, cek saldo }
        cekSaldo(rek)
        if (prompt() == 0) then
            break
        endif
    else if (me == 2) then
        { Pilihan menu kedua, tarik saldo }
        input(jumlahTarik)
        tarikSaldo(rek, jumlahTarik)
        if (prompt() == 0) then
            break
        endif
    else if (me == 3) then
        { Pilihan menu ketiga, deposit tunai }
        input(jumlahDepo)
        deposit(rek, jumlahDepo)
        if (prompt() == 0) then
            break
        endif
    else if (me == 4) then
        { Pilihan menu keempat, Transfer }
        input(rekTransfer)
        input(jumlahTransfer)
        transfer(rek, jumlahTransfer, rekTransfer)
        if (prompt() == 0) then
            break
        endif
    elif (me == 5) then
        { Pilihan menu kelima, ganti PIN }
        input(pinbaru)
        gantiPIN(rek, pinbaru)
        if (prompt() == 0) then
            break
        endif
    else
        { Bila nomor menu tidak valid }
        output("Nomor menu tidak valid.")
        if (prompt() == 0) then
            break
        endif
    end
end

```

```
endif
endwhile
```

1.3 Tugas 3

1.3.1 Pendekatan

Dalam pembuatan program simulasi ATM ini, kami menggunakan 3 file:

- ## 1. Data

Memerlukan suatu database untuk menyimpan rekening, saldo, serta PIN masing-masing rekening. Untuk ini kita menggunakan file CSV dengan bantuan dari library pandas untuk menghubungkan database dan program. File ini dinamakan data.csv.

- ## 2. Program utama

Program utama yang akan memanggil sub program lainnya. File ini dinamakan main.py.

- ### 3. Kumpulan fungsi

Menyimpan fungsi dan prosedur yang digunakan dan akan dipanggil pada main.py. File ini dinamakan fungsiatm.py.

1.3.2 Source code program

Dapat dilihat di <https://github.com/Treoxer/Tubes-ATM>

1.3.3 Screenshot Program

[illegible]

II. KESIMPULAN DAN LESSON LEARNED

2.1 Kesimpulan

Untuk membuat simulasi program ATM, pertama-tama kita perlu mendekomposisikan terlebih dahulu unsur-unsur yang ada di dalam ATM. Terdapat beberapa unsur dalam ATM yang dapat disimulasikan dalam program ATM, diantaranya cek saldo, tarik saldo, deposit saldo, transfer rekening, dan ganti pin.

Implementasi 4 pilar dari *computational thinking* (*decomposition, pattern recognition, abstraction, and algorithm design*) sangat diperlukan dalam pembuatan sebuah program. Dengan mendekomposisi masalah yang ada, kita dapat mentranslasikan bentuk pemecahan masalah ke dalam program berbasis Python secara sistematis dan solutif.

2.2 Lesson Learned

Dari pembuatan program ATM ini, kita bisa melihat bahwa dalam suatu aplikasi banyak unsur-unsur yang digunakan. Di dalam program yang cukup besar, akan diperlukan gabungan-gabungan program seperti Loop, Array, percabangan (If-Else), dan lainnya. Selain itu, program yang cukup besar tersebut akan lebih baik didekomposisi sehingga mempermudah pengerjaan program itu sendiri dengan membagi-bagi menjadi sub-sub persoalan yang pada akhirnya digabung menjadi suatu program utuh.

Pembagian program-program juga berguna ketika membaca source code sehingga lebih mudah terbaca dan dimengerti. Selain itu, data yang digunakan harus dipastikan tipenya sehingga mempermudah pengerjaan sehingga mempermudah proses “*debugging*”.

Penggunaan komentar juga sangat penting sehingga mempermudah ketika bekerja dengan orang lain. Penambahan komentar dapat menyampaikan apa maksud dari suatu baris program atau sub program yang dikerjakan sehingga “*workflow*” menjadi lebih lancar.

III. PEMBAGIAN TUGAS DALAM KELOMPOK

Berikut pembagian tugas utama yang telah kami lakukan secara inisiatif sendiri.

Gagas Praharsa Bahar (16520289) : Source Code, Video

Malik Akbar H. R. (16520299) : Dekomposisi, Laporan Tugas

Alifia Rahmah (16520309) : Pseudocode, Slide Presentasi

Ng Kyle (16520319) : Flowchart, Laporan Tugas

Selain itu, kami juga saling membantu satu sama lain dalam mengerjakan bagian masing-masing.

IV. DAFTAR REFERENSI

What is ATM Processing - How ATM Processing Works - America's ATM. (2017, February 15). Retrieved November 16, 2020, from <https://americanatm.com/atm-processing/>